

AD-A078 841

CALIFORNIA UNIV DAVIS SIGNAL AND IMAGE PROCESSING LAB F/O 9/4
 INTERFRAME ADAPTIVE DATA COMPRESSION TECHNIQUES FOR IMAGES.(U)
 AUG 79 J R JAIN & A K JAIN DAA629-78-4-0206

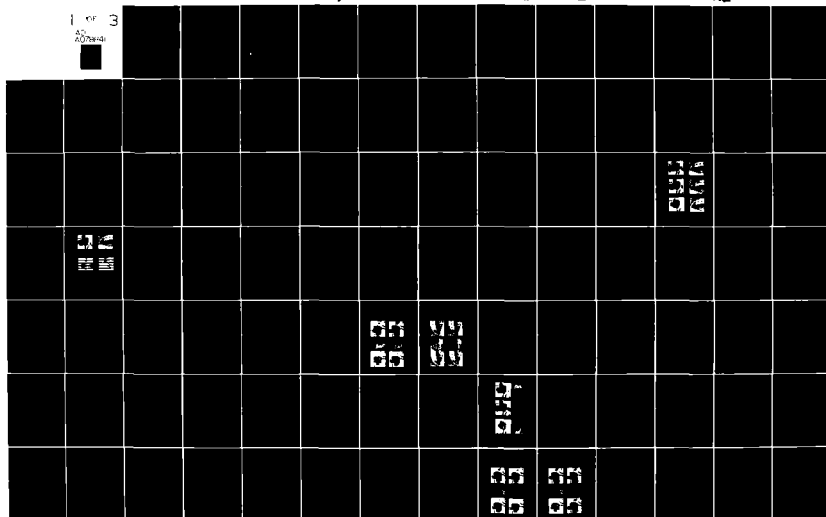
UNCLASSIFIED

ARO-16222.2-21

NL

1 of 3

AD-7841



9. SUMMARY CONTINUED

show that measured statistics in transform domain result in 2 to 4 dB improvement over commonly used separable covariance models. Interframe transform and hybrid coding schemes are compared against intraframe transform coding and some simple interframe predictive coding schemes. Also, distortion-rate curves for hybrid coding, based on models of interframe motion have been plotted. The applications of transform and hybrid coding to biomedical x-ray images has been considered. A method for joint optimization of source coding and channel coding for PCM transmission over noisy channels is presented. The performance of several transforms has been compared for some commonly used intraframe nonseparable covariance models.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ADA 078841

INTERFRAME ADAPTIVE DATA COMPRESSION TECHNIQUES FOR IMAGES*

by

Jaswant R. Jain**

Anil K. Jain**

Signal and Image Processing Laboratory

Department of Electrical and Computer Engineering

University of California

Davis, California 95616

DDC FILE COPY

August, 1979

* Research Supported in part by the U.S. Army Research Office, Durham, North Carolina, under grant DAAG29-78-G-0206, and in part by the Naval Ocean Systems Center, San Diego, California, under contract N66001-78C0151MJE. *Ny State Univ*

** J.R. Jain and A.K. Jain were with the Department of Electrical Engineering, SUNY Buffalo. J.R. Jain is now with Systems Control, Inc., 1801 Page Mill Rd., Palo, Alto, California, 94304.

79-12 27 129

The contents of this report are also
being submitted by Jaswant R. Jain to the
Graduate School of the State University of
New York at Buffalo towards his Ph.D. degree.

Accession For	
MRIS - C-441	<input checked="checked" type="checkbox"/>
EDS - 243	<input type="checkbox"/>
Unrecovered	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability/	
Dist	Available/ or special
A	

ACKNOWLEDGMENTS

The authors would like to thank Dr. Richard A. Robb of the Biodynamics Research Unit, Mayo Clinic, for his collaboration in the work on data compression of biomedical x-ray images. We would also like to thank Professor Gabor T. Herman, Department of Computer Science, SUNY Buffalo, for making the facilities of his Image Processing Laboratory available to us.

Thanks are also due to Professors V.R. Algazi and Bill Gardner, both of the Department of Electrical and Computer Engineering, University of California, Davis, Professor Fabio Rocca of Politecnico di Milano, Italy, Dr. Yuichi Ninomiya of Japan Broadcasting Corporation, Japan, Dr. Barry K. Gilbert of Biodynamics Research Unit, Mayo Clinic, and others for helpful discussions, and Mr. Shenq-Huey Wang for sharing his results and computer programs with us.

Finally, we would like to thank Nora Rogers for excellent typing and Mr. Surendra Ranganath for proofreading.

TABLE OF CONTENTS

	Page
ACKNOWLEDGMENTS	iii
LIST OF FIGURES	vii
LIST OF TABLES	xi
ABSTRACT	xiv
I INTRODUCTION	1
1.1 Interframe Data Compression Problem	1
1.2 Digital Image Transmission System and Applications of Data Compression	3
1.3 Background and Review of Current Multiframe Coding Techniques	6
1.3.1 Predictive Coding Techniques	8
1.3.2 Transform Coding Techniques	15
1.3.3 Hybrid Coding Techniques	17
1.4 Research Objectives	18
1.5 Description of Experimental Data Sets	19
1.6 Dissertation Organization	20
II MODELING, MEASUREMENT, AND ANALYSIS OF TEMPORAL CHARAC- TERISTICS	23
2.1 Motion Characteristics from Temporal Cross-Sections	23
2.2 Interframe Motion Trajectory Estimation and Modeling	26
2.3 Motion Measurement Techniques	30
2.4 Motion Measurement Results	36
2.5 Frame Repetition and Interpolation Along Motion Trajectory	43

	Page
III INTERFRAME PREDICTIVE CODING	48
3.1 Frame Replenishment with Cluster Coding	48
3.2 Adaptive Classification Prediction Coding	51
3.2.1 Motion Predictor	51
3.2.2 Intensity Prediction	54
3.2.3 Subsampling	55
3.2.4 Quantization and Coding	55
3.2.5 Buffer Length Control	55
3.2.6 Simulation Parameters	55
3.3 Results and Comparisons	58
IV TRANSFORM CODING TECHNIQUES	62
4.1 Adaptive Interframe Transform Coding Schemes	68
4.2 Experimental Results	70
4.2.1 Head and Shoulders Images	71
4.2.2 Chemical Plant Images	82
4.2.3 X-ray Projection Images	83
V INTERFRAME HYBRID CODING SCHEMES	91
5.1 Adaptive Interframe Hybrid Coding Scheme	95
5.2 Hybrid Coding with Motion Compensation	95
5.3 Distortion-Rate Curves from Models of Interframe Motion	96
5.4 Experimental Results	99
5.4.1 Head and Shoulders Images	99
5.4.2 Chemical Plant Images	109
5.4.3 Angiocardigram Images	111

	Page
VI DATA COMPRESSION FOR NOISY CHANNELS	117
6.1 Channel Encoding-Decoding of a Random Variable with MSE Criterion	118
6.2 Coding of a Random Process for Noisy Channels	126
6.3 Experimental Results and Distortion-Rate Functions	129
VII SUMMARY, CONCLUSIONS, AND SUGGESTIONS FOR FUTURE WORK	148
7.1 Summary	148
7.2 Conclusions and Suggestions for Future Work	151
APPENDICES	
A MODELING OF INTRAFRAME IMAGE STATISTICS	154
A.1 Covariance Models for 2-D Images	154
A.2 Computation of Transform Coefficient Variances	156
B COMPARISONS OF 2-D TRANSFORMS	161
B.1 Transforms and Their Performance Measure	162
B.2 Experimental Results	164
REFERENCES	171
VITA	178

LIST OF FIGURES

Figure		Page
1-1:	An Overview for Digital Transmission or Storage Retrieval	4
1-2:	Coefficients of the Seven Point Predictor Described in [10]	13
*1-3:	Some Frames of the Original Head and Shoulders and Chemical Plant Data Sets	21
*2-1:	Temporal Cross-Sections of 16 Frames of Head and Shoulders and Chemical Plant Images.	25
2-2:	Concept of Piecewise Trajectory Estimation	28
2-3:	A 2-D Logarithmic Search Procedure for the Direction of Minimum Distortion	35
2-4:	Interframe Variance as a Function of Uniform Linear Translation (i,j) Measured over One Complete Frame	37
2-5:	Results of Motion Measurement Relative to the Previous Frame on a Portion of Head and Shoulders Frame No. 8 for Sub-block Size of 16×16	39
2-6:	Results of Motion Measurement Relative to the Previous Frame on a Portion of Chemical Plant Frame No. 12 for Sub-block Size of 16×16	40
*2-7:	Effects of Motion Compensation on Interframe Prediction and Interpolation of Head and Shoulders Frame 8	44
*2-8:	Effects of Motion Compensation on Interframe Prediction and Interpolation of Chemical Plant Frame 12	45
3-1:	Buffer Control Levels for Frame Replenishment Cluster Coding	50
3-2:	Nearest Neighbors in the Present Frame for Adaptive Classification Prediction Scheme	50
3-3:	Nearest Neighbors in the Previous Frame for Adaptive Classification Prediction Scheme	50
3-4:	An Adaptive Classification Prediction Scheme	52

* These figures contain images.

Figure	Page
3-5: Buffer Control Levels for the Adaptive Classification Prediction Scheme	52
*3-6: Results of Interframe Predictive Schemes for Head and Shoulders Frame 8	60
4-1: An Interframe Transform Coding Scheme	63
*4-2: Intraframe Transform Coding using Separable and Isotropic Models, Sub-block Size = 16×16	74
*4-3: Intraframe Transform Coding using Measured Statistics in Transform Domain with Various Sub-block Sizes	75
*4-4: Interframe Transform Coding, Sub-block Size = $16 \times 16 \times 16$	80
*4-5: Intraframe and Interframe Transform Coding using Measured Statistics in Transform Domain	85
*4-6: Images Resulting from Data Compression of a Projection Image at Angle of View = 0°	86
4-7: Variation of Mean Square Error as a Function of Frame Number for Transform Coding of the X-ray Projection Images	87
*4-8: Reconstruction Images	89
5-1: A Non-Adaptive Interframe Hybrid Coding Scheme	92
5-2: An Adaptive Interframe Hybrid Coding Scheme	92
5-3: Distortion-Rate Curves for Hybrid Interframe Coding of Images with Isotropic Intraframe Covariance and for Various Distributions of Interframe Motion Uncertainty	98
5-4: Histogram of the Activity Index (IFV) of Head and Shoulders Images. Sub-Block Size = 16×16	102
5-5: Signal to Noise Ratio as a Function of Frame Number for Coding of the Head and Shoulders Images	103
5-6: Bit-Rate as a Function of Frame Number for Adaptive Hybrid Coding of the Head and Shoulders Image	103
*5-7: Interframe Hybrid Coding using Isotropic Model with Correction, Sub-block Size = 16×16	105
*5-8: Interframe Adaptive Hybrid Coding with Motion Compensation	110

Figure	Page
5-9: Bit Allocation for 16×16 Sub-block Size for Angiocardiogram Images	114
5-10: Classification Maps for Two Frames of the Angiocardiogram Images. Average Bit-Rate = .125	114
*5-11: Images Resulting from Data Compression of Angiocardiogram Images	116
6-1: PCM Transmission of a Continuous Random Variable having Zero Mean and Unit Variance over a Noisy Channel	119
6-2: A Data Compression Scheme for Noisy Channels	127
6-3: Total Distortion as a Function of Quantization Bits. $n = 8$	138
6-4: Distortion-Rate Curves for PCM Transmission of a Random Variable over a Binary Symmetric Channel with Channel Optimization	139
6-5: Distortion-Rate Curves for a One-Dimensional First Order Markov Process with Gaussian Distribution Using Channel Optimization. $\rho = .95$	140
6-6: Distortion-Rate Curves for a 2-D Random Process with Gaussian Distribution using Channel Optimization. $\rho_i = \rho_j = .95$	140
6-7: Bit-Assignment for 16×16 Cosine Transform Coding for a 2-D Isotropic Random Field with $\rho_i = \rho_j = .95$. Bit-Rate = 1 Bit/Sample	141
6-8: Percentage of Redundant Bits Assigned for Channel Error Correction for Coding of a 2-D Isotropic Random Process with Gaussian Distribution using Channel Optimization. $\rho_i = \rho_j = .95$	142
6-9: Distortion-Rate Curves for 16×16 Cosine Transform Coding of a 2-D Isotropic Random Field for Various Schemes. $\rho_i = \rho_j = .95$	142
*6-10: Images Resulting from 16×16 Cosine Transform Coding and Transmission over a Binary Symmetric Channel	145
*6-11: Error Images Corresponding to the Coded Images of the Previous Figure	146

Figure		Page
B-1:	Basis Restriction Error of an 8×8 Transform for the Isotropic Covariance Model. $\rho_i = \rho_j = .95$	166
B-2:	Basis Restriction Error of an 8×8 Transform for the NC-1 Model	166
B-3:	Basis Restriction Error of an 8×8 Transform for the Measured Covariance Model for a Girl Image	167
B-4:	Distortion-Rate Curves for Transform Coding of an 8×8 Array for the Isotropic Covariance Model with Gaussian Distribution. $\rho_i = \rho_j = .95$	167
B-5:	Distortion-Rate Curves for Transform Coding of a 16×16 Array for the Separable Covariance Model with Gaussian Distribution. $\rho_i = \rho_j = .95$	168
B-6:	Distortion-Rate Curves for Transform Coding of a 16×16 Array for the Isotropic Covariance Model with Gaussian Distribution. $\rho_i = \rho_j = .95$	168

LIST OF TABLES

Table	Page
2-1: Improvement in SNR of Interframe Prediction Error (IFPE) Due to Motion Compensation	42
2-2: Improvement in SNR of Interpolated Frame (From the Preceding and the Following Frames of the Original Data) Due to Motion Compensation	42
3-1: Quantization for Adaptive Classification Prediction Scheme	57
3-2: Frame Replenishment Cluster Coding Results for Head and Shoulders Data	59
3-3: Adaptive Classification Prediction Coding of Head and Shoulders Data, Bit Rate = 0.5	59
4-1: SNR for Non-Adaptive <u>Intraframe</u> Cosine Transform Coding of the Head and Shoulders Images for Three Statistical Models. Sub-block Size = 16×16	73
4-2: SNR for Non-Adaptive <u>Intraframe</u> Cosine Transform Coding of Head and Shoulders Images for Two Statistical Models. Sub-block Size = 64×64	73
4-3: SNR for Non-Adaptive <u>Interframe</u> Cosine Transform Coding of the Head and Shoulders Images for Two Statistical Models. Sub-block Size = $16 \times 16 \times 16$	73
4-4: Bit Allocation for the First Eight Indices along Temporal Axis Corresponding to the 3-D Separable Covariance Model with $\rho_i = \rho_j = \rho_k = .95$ for Non-Adaptive Cosine Transform Coding. Sub-block Size = $16 \times 16 \times 16$, Bit-Rate = .5 Bit/Pixel	77
4-5: Bit Allocation for the First Eight Indices along Temporal Axis Corresponding to the Measured Statistics for the Head and Shoulders Data for Non-Adaptive Cosine Transform Coding. Sub-block Size = $16 \times 16 \times 16$, Bit-Rate = .5 Bits/Pixel	78
4-6: Parameters of 4-Class Adaptive Transform Coding Scheme for Head and Shoulders Data	81

Table	Page
4-7: Performance of Adaptive Transform Coder for Head and Shoulders Data. <u>Same Statistics</u> were used for all the Classes and were Measured over all the Data. Block Size is $16 \times 16 \times 16$	81
4-8: Performance of Adaptive Transform Coder for Head and Shoulders Data. <u>Different Statistics</u> were used for Different Classes and were Measured over each Class. Block Size is $16 \times 16 \times 16$	81
4-9: SNR for Non-Adaptive Intraframe and Interframe Cosine Transform Coding of the Chemical Plant Images with Measured Statistics	84
4-10: Performance of the Non-Adaptive Interframe Transform Coder for the X-ray Projection Images	84
5-1: Comparisons of Rates for Interframe Hybrid Coding for Various Distributions of Motion Uncertainty	99
5-2: Performance of the Non-Adaptive Hybrid Coding Scheme for the Head and Shoulders Data	101
5-3: Parameters of the 4 Class Adaptive Hybrid Scheme for Head and Shoulders Data	101
5-4: Performance of the Adaptive Hybrid Coding Scheme for the Head and Shoulders Data	101
5-5: Parameters of the 4 Class Adaptive Hybrid Coding Scheme with Motion Compensation for Head and Shoulders Data	108
5-6: Parameters of the 4 Class Adaptive Hybrid Coding Scheme with Motion Compensation, using Alternate Frame Skipping and Interpolation, for Head and Shoulders Data	108
5-7: Performance of the Adaptive Hybrid Coding with Motion Compensation for Head and Shoulders Frames 5 thru 9. Skipped Frames are Interpolated along the Motion Trajectory	112
5-8: Parameters of the 4 Class Adaptive Hybrid Coding with Motion Compensation for the Chemical Plant Images	112
5-9: Performance of the Adaptive Hybrid Coding with Motion Compensation for the Chemical Plant Images. Sub-block Size = 16×16	112
5-10: Parameters of a 4 Class Adaptive Hybrid Coding Scheme for the Angiocardiogram Images	115

Table	Page
5-11: Performance of the Adaptive Hybrid Coding Scheme for the Angiocardiogram Images. <u>Same</u> Spatial Measured Statistics were used for all Classes. Sub-block Size = 16×16	115
6-1: Quantizer Distortion $q(k)$ for Various Densities	130
6-2: Basis Vectors $\{g_i; i=1, \dots, k\}$ of Group G for (n,k) Group Code.	130
6-3: Channel Distortion for (n,k) Block Coding of a Unit Variance (at the Input of the Quantizer) <u>Gaussian</u> Random Variable	132
6-4: Channel Plus Quantizer Distortion for (n,k) Block Coding of a Unit Variance (at the Input of the Quantizer) <u>Gaussian</u> Random Variable	133
6-5: Channel Plus Quantizer Distortion for (n,k) Block Coding of a Unit Variance (at the Input of the Quantizer) <u>Uniform</u> Random Variable	134
6-6: Channel Plus Quantizer Distortion for (n,k) Block Coding of a Unit Variance (at the Input of the Quantizer) <u>Laplacian</u> Random Variable	135
6-7: Effect of Encoder Mapping β	136
6-8: Channel Distortion for $(8,6)$ Encoder	138
6-9: Performance of Data Compression Schemes at 1 Bit/Pixel for Cosine Transform Coding of Girl Image	144
A-1: Transform Coefficient Variances for 16×16 Sub-block Size Cosine Transform Measured over the 16 Frames of Head and Shoulders Data	159
A-2: 16×16 Covariance Matrix, R, for a Two-Dimensional Isotropic Stationary Random Field for $\rho_i = .955$, $\rho_j = .945$	159
A-3: 16×16 Cosine Transform Coefficient Variances Corresponding to the Covariance Matrix of Table A-2	160
A-4: Transform Coefficient Variances of the Above Table after Correction Factor of Eqn. (A-10)	160
B-1: Distortion-Rate Comparisons of Various Discrete Transforms for an Isotropic Covariance Field with $\rho_i = \rho_j = 0.95$, Unit Variance and Gaussian Distribution	169

ABSTRACT

A method of interframe motion measurement and compensation, based on approximation of piecewise linear translation of small rectangular areas, is presented. This method significantly improves the temporal correlation (10 to 12 dB reduction in interframe variance) and permits reduction of sampling rate along temporal axis. A technique of visual characterization of interframe characteristics, based on temporal cross-sections, is described.

Some commonly used statistical models for prediction of the variances of transform coefficients, in intraframe and interframe transform coding, are compared. Results show that measured statistics in transform domain result in 2 to 4 dB improvement over commonly used separable covariance models. A method of adaptation for the local changes in image statistics in transform and hybrid coding is developed. This results in great improvement (about 4 dB) in performance. A hybrid coding scheme using motion compensation, frame skipping, and interpolation of skipped frame along motion trajectory is presented, which further improves the coder performance.

Interframe transform and hybrid coding schemes are compared against intraframe transform coding and some simple interframe predictive coding schemes. Also, distortion-rate curves for hybrid coding, based on models of interframe motion have been plotted. The applications of transform and hybrid coding to biomedical x-ray images has been considered, and the results show that significant compression can be realized for these images. The effects of distortion due to data compression of x-ray

projection images (used in computed tomography) on the reconstruction images (by inverse Radon transform) have been evaluated.

A method for joint optimization of source coding and channel coding for PCM transmission over noisy channels is presented. It is shown how this method can be applied to transform coding of images. The results show that this method performs significantly better than the conventional error correcting codes or schemes with no channel protection. At a rate of 1 bit/pixel and channel error probability of 10^{-2} , the proposed method results in 10 dB improvement over an ordinary transform coder.

The performance of several transforms has been compared for some commonly used intraframe nonseparable covariance models. The results indicate that the cosine transform performs very close (.05 dB at 1 bit/pixel) to the optimum K-L transform.

CHAPTER I

INTRODUCTION

A monochrome image* is a function of two spatial variables. Many imaging systems generate multiple frames of images. These multiple frames could be a function of time (e.g., in television) or other variables (e.g., angle of view and time in dynamic spatial reconstruction, Robb [63]). Sampling of these variables depends on the application, e.g., in broadcast television the time axis is sampled at 50-60 samples per second to avoid flicker.

For digital processing, an image frame is sampled along both the spatial axes. Nyquist sampling theory provides the most important step towards a reduction of digital information required to represent a continuous signal. This theory states that any bandlimited signal, sampled at a rate greater than twice its highest frequency content, could be reproduced without introducing any distortion. In the simplest binary coding, each sample of an image, called a pixel, is quantized independently by a finite number of bits. This is called pulse code modulation (PCM). For raw image data, each pixel is uniformly quantized, and is represented by a fixed number of bits. For human viewing of an image, 8 bits/pixel gives sufficient resolution. For broadcast television, the data rate for PCM transmission of images is approximately 65 Mbits/sec.

1.1 Interframe Data Compression Problem

It is evident that the enormous data rates generated by multiple frame images would result in high costs of transmission and/or storage.

* as opposed to a color image

Thus, there is a great need for reducing the data rates as much as possible. Interlacing of the fields in television broadcasting is a form of data compression which exploits the retention properties of human vision.

A simple statistical or visual analysis of the image data reveals that there is very high correlation between adjacent pixels, both within a single frame and from frame to frame. This high correlation results in significant redundant information in the original raw data. The basic problem of data compression is to effectively exploit this redundancy to reduce the data rates.

A number of data compression (also called coding) schemes have been developed for single image frames, [5,22,29,31,36,38,51,53,58,75,77,83,87]. These are called intraframe coding schemes and are based on exploitation of spatial redundancy. The interframe coding schemes on the other hand, utilize the redundancy between the frames as well as within the frames and generally achieve higher compression than the intraframe schemes.

In principle, it is possible to compress the digitized data without introducing any further distortion (digitization itself introduces distortion). However, such schemes do not yield large enough compression ratios. It is possible to achieve much larger compression by introducing small but acceptable (depending on the application) distortion in the originally digitized data. Thus, we need some quantitative and qualitative measures of the distortion. The problem of data compression then becomes the minimization of data rates for a given distortion level or, equivalently, minimization of the distortion for a given data rate.

The quantitative distortion measure we use is the well known mean square error (MSE or m.s.e.) criterion. Let $u_{k,i,j}$ be the intensity of

a sample of a digitized three dimensional image data array and $u_{k,i,j}^*$ be its reproduced value after data compression. Then the MSE due to data compression is defined as

$$MSE = \frac{1}{N_o} \sum_i \sum_j \sum_k (u_{k,i,j} - u_{k,i,j}^*)^2$$

where N_o is the total number of samples in the array over which the MSE is being measured. Such a global criterion of overall mean square error is not always very meaningful, especially at moderate to high levels of distortions. So this has to be used with some qualitative measures to judge the quality of the reproduced images. Some qualitative measures are given in [12]. A simple method is to judge the images by viewing the encoded image and comparing it with the original image. The inspection of the error images (amplified to give full dynamic range) is also very informative about the distribution and structure of the errors. Sometimes the MSE measured over locally homogeneous regions of an image is also quite useful. More sophisticated criteria, such as frequency weighted mean square error [53], or visibility of errors, etc., are possible, but are difficult to incorporate in interframe data compression algorithms.

The MSE is also expressed by a quantity called the signal to noise ratio (SNR or S/N), defined in decibels (dB), as

$$SNR = 10 \log_{10} \left\{ \frac{(\text{Peak to Peak Signal})^2}{MSE} \right\} \text{ dB} .$$

1.2 Digital Image Transmission System and Applications of Data Compression

Figure 1-1 shows a schematic of a typical digital image transmission (or storage-retrieval) system. Block 1 consists of an image sensing or

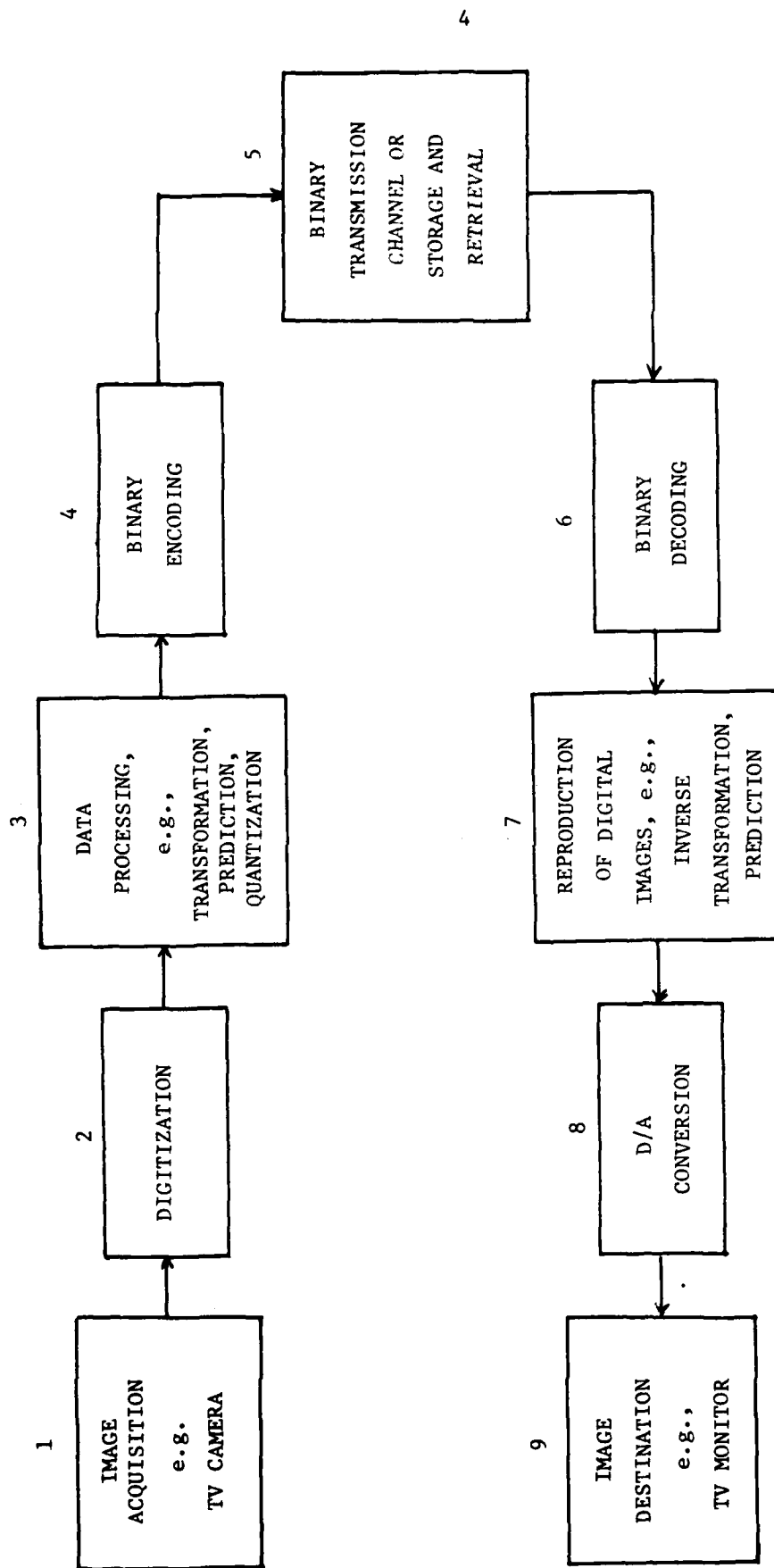


Figure 1-1: An Overview for Digital Transmission or Storage Retrieval System for Images

acquisition system. It could consist of a continuous sensor (e.g., a raster scanning camera) or an array of detectors arranged at the sampling grid. The acquired image signal is then sampled (if continuous) and digitized in block 2. Block 3, which is of most interest to us, contains a processor which performs data compression. The compressed data is encoded into bits in block 4. Error protection for transmission over noisy channels is also done here. The binary coded data is transmitted or stored (block 5). Blocks 6 through 9 perform the inverse of most of the functions performed in blocks 4 through 1 (not all the functions performed in these blocks are invertible, e.g., quantization). For the most part (chapters II to V) we will be concerned with blocks 3 and 7. There, we have integrated blocks 4-6 into a single block named "channel". In chapter VI, where we deal with image coding for noisy transmission channels, we will consider blocks 4-6 in detail.

There are several considerations in developing the data compression algorithms of block 3 in addition to reducing the data rate. These considerations include the complexity of binary encoder and decoder, real time processing in blocks 3 and/or 7, uniform or variable data rates, amount of storage, size of channel buffer, noise characteristics of the channel, etc.

There are many applications where interframe data compression of images could be used with great savings in transmission/storage costs. These include television transmission between stations, teleconferencing, videotelephone, satellite images, biomedical x-ray images in computer aided tomography and angiocardiology, etc. For future projections for satellite communication traffic for television and teleconferencing, see [73].

1.3 Background and Review of Current Multiframe Coding Techniques

The multiframe data compression schemes reported in the literature have been applied mostly to video images, generated by a television or a movie camera, where the interframe variable is time. In some applications, e.g., television and videophone, a time evolving scene is registered as a sequence of equi-interval images by a camera, which is mostly stationary. In other applications, such as remotely piloted vehicle (RPV), a moving camera is capturing a time changing scene. In the former applications, the changes from one frame to the next, with respect to a fixed location in the frame, are mostly localized in some areas of the frame. While in the latter case, such changes occur throughout the frame. These changes, including those due to zooming the camera, are called inter-frame motion.

A simple method of detecting motion between two consecutive image frames is by measuring the temporal changes between them. Suppose $u_{k,i,j}$ represents the intensity of the (i,j) th pixel of the k th frame. If the interframe difference (IFD) signal,

$$d_{k,i,j} = u_{k,i,j} - u_{k-1,i,j} ,$$

exceeds a certain threshold, then the (i,j) th pixel of the k th frame is classified as moving. An inherent assumption here is that the illumination from one frame to the next remains unchanged. Generally, the moving pixels occur in clusters [13], and constitute the so called moving areas. The rest of the image constitutes the stationary areas. Techniques which are adjusted according to the local changes in spatial and temporal characteristics of images are called adaptive methods.

Interframe coding techniques could be broadly classified into three categories. The first category, a subject most investigated, is called predictive coding. Here the intensity value of a current pixel in a raster scanned image is predicted from the knowledge of the previously scanned and coded pixels. Generally, the pixel neighborhood used in prediction is limited to be a small set of pixels in the present and the preceding image frames. This is because of the Markovian nature of the data and it limits the memory requirements to slightly more than one image frame. The prediction error, which represents the new information in the current pixel, is quantized and coded. For highly correlated data, the prediction error is generally small and can be coded by much fewer bits than required in PCM transmission.

The second category, developed more recently for multiframe images, is called transform coding. While in most predictive coding schemes we end up with as many samples as the input data, transform coding packs the information in much fewer samples which need to be coded. Typically, the interframe data is divided into smaller three dimensional arrays, called sub-blocks, of equal size. Each sub-block is then operated upon by a three dimensional, separable, unitary transform and the selected transformed samples are quantized and coded independently. The sub-blocks are reconstructed by taking the inverse transform. This method requires storage equal to the number of frames in the temporal direction of the sub-blocks.

The third category is a combination of the above two and is called hybrid or transform/predictive coding. Here, each image frame is divided

into equal size sub-blocks and each sub-block is transformed by a two dimensional separable unitary transform. Predictive coding is then performed along the temporal axis, for each transformed sample, to exploit frame to frame correlation.

Interframe coding techniques have gained momentum only since the mid-sixties. A brief review of the recent work is given below for each of the three categories.

1.3.1 Predictive Coding Techniques - Due to their simple hardware realization, considerable work has been done on predictive coding schemes. In [52], prediction is based on the previous pixel of the same scan line and the technique is essentially an intraframe one dimensional DPCM.

Recognition of the fact that a vast majority of pixels in a given frame do not differ noticeably from the corresponding pixels of their preceding frame (i.e., most of the image field in successive frames is stationary) has led to interframe predictive schemes which do not require transmission of stationary pixels. In [48], the prediction of a pixel is simply the intensity value of the corresponding pixel in the preceding frame. If the absolute value of the prediction error is larger than a threshold, it is quantized and coded together with the address of the pixel. Otherwise, the value of the pixel in the preceding frame is repeated. This technique is called conditional replenishment because only the moving areas of the image are replenished from one frame to the next. It is evident that the rate at which the code is generated varies depending on the size of and activity in the moving areas. If the transmission channel is designed for an average data rate, then an arbitrarily large buffer would be required to take care of large fluctuations in the level

of motion. To limit the buffer requirement to a reasonable size, a variable threshold is used. The threshold is increased as the buffer is filled up, thereby reducing the rate at which the data is generated.

It has been noted [57] that the spatial resolution in the moving areas and the temporal resolution in the stationary areas of an image can be reduced without noticeable reduction in the quality of perception of the scene. This is called the exchange of spatial and temporal resolution. In [44] a simple coder has been described which exploits this exchange of resolution to reduce the data rate.

In [13], the conditional replenishment method of [48] has been improved, and some of the techniques of [44], together with some other adaptations, have been used to result in a more efficient coder with increased complexity. We have simulated this technique for comparison purposes and a brief description is given in chapter III. A review of the above techniques and some other simpler techniques is given in [24].

If some area of a frame is moving at a speed larger than 1 pixel/frame, then it is obvious that for a pixel belonging to such an area the correlation with intraframe neighbors would be higher than that with the corresponding pixel in the preceding frame. Hence, in the absence of the knowledge about the direction of the motion, an intraframe prediction error would have a lower variance than the interframe difference signal. Thus, more compression can be achieved by coding the intraframe prediction error in such moving areas. A scheme utilizing this is reported in [42]. To detect the moving areas an adaptation, better than that of [48] and [13], is used.

The correlation, power spectrum, and some other properties of frame-difference signal are reported in [17]. Mathematical analysis as well as experimental results are given. The results of this study have been used in [42] to design a better segmenter of moving and stationary areas for the purpose of coding. The simulation results for the entropy of prediction error signal for a variety of predictors (using different combinations of the neighborhood pixels in the present and the preceding frame) at various speeds are reported in [23]. Pictures with different resolutions have been used to compare entropy versus resolution at various speeds.

Most of the coders described in [13,24,42,44] are designed for a data rate of 0.25 - 1.0 bits/pixel and for a signal with 1 MHz bandwidth, the data rate is .5 - 2 Mbits/sec. In [26] a very low bit rate coder, .1 bit/pixel, or 0.2 Mbits/sec., has been described which reproduces the stationary areas quite well, but scenes containing moderate and large motions are visibly smeared and blurred. It combines cluster coding of [13], a higher order prediction given by the line-to-line difference of the frame difference signal, subsampling in spatial and temporal directions as needed, temporal filtering, etc., to achieve this low data rate. Low pass temporal filtering of the signal is done to reduce the entropy of the prediction error. A near-ideal low pass filter would require several frame memories. To limit the memory requirement to one frame, a simple temporal filtering could be performed by sub-sampling in temporal direction and then interpolating the missing frames. In [26] temporal filtering is performed by a simple averaging of the incoming frame and the previously stored frame. Because of the temporal filtering by averaging, the

jerkiness in motion due to temporal subsampling is less objectionable.

Most of the experiments reported in the aforementioned literature were carried out on data sampled at about 2×10^6 samples/sec. The coder of [13] at 1 bit/pixel thus has a data rate of 2 Mbits/sec. For many applications, a higher resolution with a sampling rate of 8×10^6 samples/sec. is desired. At 1 bit/pixel it would require a high data rate of 8 Mbits/sec. A coder is described in [25], which compresses the data rate to 1.5 Mbits/sec. or .19 bit/pixel. It is reported to give acceptable quality with some blurring of the moving areas in TV-conference type of applications, where for the most part the camera is stationary and the moving subjects do not move too rapidly. This coder utilizes conditional replenishment of [13], moving area segmenter of [42], a higher order predictor, and a temporal filter, as in [26]. Variable quantizer levels and sub-sampling in spatial domain are used to maintain a smooth data rate.

A coder which uses the interframe sample difference, temporal filtering by attenuating the frame difference signal, spatial subsampling in both directions when buffer starts filling up, variable length code words, etc., is reported in [84]. It is designed for 4 MHz videotelephone and NTSC color TV signals, and operates at an average rate of 6.312 Mbits/sec. An interframe coder for NTSC color TV signals has also been built by Nippon Electric in Japan and has been reported in [30]. This is designed for high quality transmission and operates at 16-32 Mbits/sec.

A higher order prediction coder which differs considerably from those discussed above is described in [10]. It assumes that the interframe data is a sample of a 3-D wide sense stationary random process whose

covariance is separable and first order Markov in each dimension. Under this assumption, the optimal predictor is based on seven pixels with prediction coefficients directly related to the one step correlation in each direction. Figure 1-2 shows the pixels and their prediction coefficients used in the prediction of the point marked S. The prediction error is quantized and coded using variable word length codes. The one step temporal coefficient, α_F , has been set equal to 1 to give better prediction and low data rate in the stationary areas. This scheme also generates data at a nonuniform rate and to keep buffer requirement reasonably low, some adaptations have been made to reduce data rate when the buffer starts filling up. Depending on the buffer contents, a temporal-spatial filtering is performed in which a weighted average of the interframe difference signals of the neighboring pixels is taken. The weights are controlled by buffer contents. An additional temporal filtering is used when buffer overflow is imminent. This is achieved by attenuating the output of the temporal-spatial filter used to reduce the entropy of the signal.

In all of the above techniques, the motion in any area of the scene is inferred from the magnitude of the interframe difference signal. No efforts are made to measure the nature and the direction of the motion. Due to computational and dimensionality problems, most of the motion analysis of interframe images has been restricted to translational motion, see e.g. [9,11,66]. In [66] a mathematical analysis is presented where an image is divided into smaller areas or zones. For each zone the displacement vector (x and y coordinates of the motion) and the corresponding prediction errors (the prediction is based on the pixel in the previous frame corresponding to the displacement vector of that zone) are transmitted. For the purpose of analysis, a mathematical model

Previous
Frame

	$+\alpha_s \alpha_L$	$-\alpha_L$	
	$-\alpha_s$	1	

Present
Frame

	$-\alpha_s \alpha_L$	$+\alpha_L$	
	$+\alpha_s$	S	

α_s - One step sample-to-sample correlation

α_L - One step line-to-line correlation in the same field

NOTE: Each frame consists of two interlaced fields

Figure 1-2: Coefficients of the Seven Point Predictor Described in [10].

for the random video process is constructed to determine the optimum size of the zone which can be represented by a single displacement vector.

Methods for measuring small displacements, and segmenting an image into stationary and moving areas with different displacement, has been considered in [11]. Based on linear regression and approximation, simple formulas are derived where the motion could be measured from interframe difference signal and first order spatial differences in x and y directions. To segment an image into moving and stationary areas, it is assumed that there is only one moving object undergoing translation. A two state Markov model with known state transition probabilities is assumed. A maximum a posteriori (MAP) detector of the Markov chain is found using the Viterbi algorithm by observing interframe differences and assuming them to be an independent sequence. Then the method is extended to more than one moving object. Displacement measurement accuracy of .1 pixel/frame for motion up to 2-3 pixels/frame has been reported.

An interframe coder using image segmentation and motion measurement is described in [9], together with some experimental results. Each frame is segmented into three areas, namely, stationary background, translating objects, and areas which cannot be predicted from the previous frame, via a tri-state MAP estimator. For stationary background and translating objects, prediction is based on the corresponding pixels from the previous frame. While for the remaining areas a spatial predictor is used.

A somewhat different approach to motion estimation and its application to interframe coding has been recently published in [50]. Here a pixel by pixel translational motion is recursively estimated and the interframe prediction is based on the estimated motion-displaced-location

in the previous field. The prediction error is cluster coded similar to [13].

1.3.2 Transform Coding Techniques - The superior performance of transform coding over other techniques for coding intraframe images is well known. Its extension to interframe coding using 3-D transforms was not attempted until recently, mainly because of the requirement of storing several frames at the transmitter as well as at the receiver, resulting in exorbitant memory costs. Recent developments in digital technology now make it possible to store several image frames and thus make transform coding feasible. Knauer [39] has reported some results on Hadamard transform coding. He considers a block of 4 image frames at a time. Each frame consists of two interlaced fields and contains 525×512 pixels, each pixel originally quantized to 6 bits. This block of 4 frames is divided into sub-blocks of size $4 \times 4 \times 4$. Each sub-block is transformed by a 3-D Hadamard transform (for definitions of various transforms used in data compression, see [3,31,58]) and the transform coefficients are truncated to 8 bits. To design the coder at a given bit-rate, a fixed number of bits are distributed among various transform coefficients, a majority of which are assigned no bits. The bit assignment has been found by trial and error to give good visual quality. The coder can adapt to motion by keeping high spatial resolution for stationary areas and exchanging it for temporal resolution in moving areas.

The transform coder of [39] lacks the mathematical analysis in bit assignment, which is an important aspect of transform coding. Roese [67], Roese, et al. [68] and Natarajan and Ahmed [49] have extended the mathematical analysis of 2-D transform coding to three dimensions and have also reported experimental results on interframe coding.

In [67,68] the interframe image random field is modeled as a 3-D, wide sense stationary first order Markov field with separable covariance function in each dimension. The interframe image data is divided into smaller 3-D sub-blocks and then transform coded independently. The bit assignment is based on the separable covariance model and Shannon rate distortion bound for the quantizer. The transform samples are quantized using a compander which performs very close to the optimum Max quantizer. The distribution for each transform sample is assumed to be Laplacian, except for the DC term, for which a Rayleigh distribution is assumed. These distributions are reported in [67] to be good approximations for image data. The variances of the transform samples are found from the covariance model chosen. Theoretical performance of the coder using Cosine transform for various block sizes has also been reported. The mean square error decreases with the increase in block size, but it also increases the complexity. The experimental results on the actual data are also reported for the Cosine transform at various bit-rates for a block size of $16 \times 16 \times 16$.

The fact that the multiframe data cannot be satisfactorily modeled by separable statistics (covariance), a model such as described in [67] could yield poor coding performance. But we do need the transform domain variances to design a coder without resorting to trial and error. In [49] the authors suggest calculating the 3-D covariance function on a portion of the image data over a window of the same size as the sub-block and assuming the random process to be wide sense stationary. From this, the transform domain variances could be calculated by appropriately taking the transform. The three dimensional sub-blocks are stored as one dimensional arrays by lexicographic ordering to facilitate the addressing. A

Kronecker product of the transform matrices is then used to find the equivalent 3-D transform of this array. The experimental results for a block size of $4 \times 4 \times 4$ for Cosine and Hadamard transforms are reported at 1 bit/pixel for 4 MHz signals.

1.3.3 Hybrid Coding Techniques - Because of difficulties in modeling interframe image fields as well as the increased complexity of the transform coders, transform/predictive or hybrid coding techniques have also been investigated in [67,68]. These are extensions of the intraframe hybrid coding described in [22]. Each frame is divided into smaller equal size sub-blocks and each sub-block is transformed by a unitary transform. Then a linear first order predictor is used in the temporal direction. In a simple or non-adaptive scheme a separable first order Markov model in each dimension is used. Based on this model the calculation of transform domain variances and the optimum prediction coefficient are easily found. Theoretical performance of this coder for various sub-block sizes is reported. Another scheme, in which the local changes in the statistics are taken into account by measuring the statistics, at the transmitter as well as the receiver (and using these statistics for coding) has been reported with many experimental results. This scheme has been called adaptive hybrid coding. Results for discrete Cosine and Fourier transforms at various bit-rates have been reported together with the effect of channel errors. The adaptive hybrid scheme shows a much better performance compared to the non-adaptive hybrid coding and transform coding schemes based on a 3-D separable model. Also, some methods and experimental results for motion compensation of the camera motion have been reported in [67].

1.4 Research Objectives:

In the broad context of interframe coding with emphasis on hybrid and transform techniques, the following problems have been identified and solutions proposed together with experimental results.

(i) The transform and hybrid coders reported so far allocate equal bits to all the areas of an image. It is intuitively obvious that the stationary areas with no interframe activity could be transmitted with little or no bits, while those with more activity would require more bits to transmit the changes. Our objective is to find ways of classifying sub-blocks of images into classes of varying temporal and spatial activity and assignment of bits for various classes.

(ii) Although, in general, the interframe motion is difficult to characterize, in most cases it could be approximated by linear translation. Since hybrid coding is based on a block by block coding, we investigate the methods of measuring translation on a block by block basis and use it for data compression. A technique for frame interpolation along motion trajectory will be investigated to achieve higher data compression.

(iii) Since the trajectory of motion of a pixel (or a block) cannot be estimated perfectly, we consider models and effects of uncertainty in trajectory estimation for data compression. We would also like to find rate-distortion curves based on such models.

(iv) We investigate the problem of joint optimization of data compression and channel encoding for minimizing the overall mean square error for image transmission over noisy channels.

(v) As pointed out earlier, the interframe data compression schemes have been applied only to video images. We look at another potential area of application, data compression of biomedical x-ray images.

(vi) The performance of the hybrid and transform coding methods is dependent on the choice of the transform. We evaluate the relative performance of various transforms for a variety of non-separable two dimensional random fields which have been used for modeling image covariance statistics. Previous results have considered only the separable covariance model.

1.5 Description of Experimental Data Sets:

We have used four very distinct types of multiframe image data sets. Two of these data sets are video motion images obtained from the Naval Ocean Systems Center, San Diego, California. The other two data sets are x-ray images obtained from the Biodynamic Research Unit, Mayo Foundation, Rochester, Minnesota. A brief description of these data is provided below.

(i) Head and Shoulders (H & S) - contains 16 sequential frames of 16 mm, 24 frames/second, motion picture of a subject (Walter Kronkite) against a stationary background in conversation, digitized to 256×256 pixels/frame, 8 bits/pixel.

(ii) Chemical Plant - an aerial view of a complex of buildings and roads from a moving platform, 16 frames digitized to 256×256 pixels/frame, 8 bits/pixel. It contains a fourth of a 35 mm frame digitized to 512×512 pixels/frame.

(iii) Angiocardiograms - 100 X-ray images of the left ventricle of a human heart taken at intervals of 1/30 sec., after injection of a contrast material in the blood, contain four complete heart beat cycles. Each frame digitized to 128×176 pixels with 8 bits/pixel. Spatial resolution is .5 mm.

(iv) Projection Images - 120 X-ray projections of a dead dog's thorax taken at intervals of 3° around an axis approximately through the center of the thorax. Each image digitized to 128×128 pixels with 8 bits/pixel. Spatial resolution is ≈ 1 mm. These images are used for 3-D reconstruction of the X-ray absorption densities of the thorax.

Since each data set is digitized to 8 bits/pixel, the intensity range of the original data is between 0 and 255. Therefore, for the calculations of SNR, the peak-to-peak value of the signal has been taken as 255, even though the actual peak-to-peak signal could be somewhat smaller.

1.6 Dissertation Organization

We first start with the problem of modeling and understanding of the temporal characteristics of the motion images in Chapter II. There we propose some methods of translational motion measurement on a block by block basis. Then we analyze the effects of uncertainty in motion estimation and define some parameters which give simple measures of this uncertainty and are useful for developing data compression algorithms. We also propose a method of data compression based on temporal subsampling and interpolation of the missing frames along the motion trajectory.

In Chapter III we describe two interframe predictive schemes. One of these schemes is reported in [13], and the other one is a simple



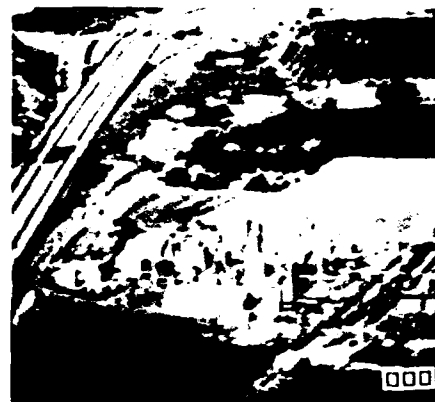
Frame No. 4



Frame No. 8



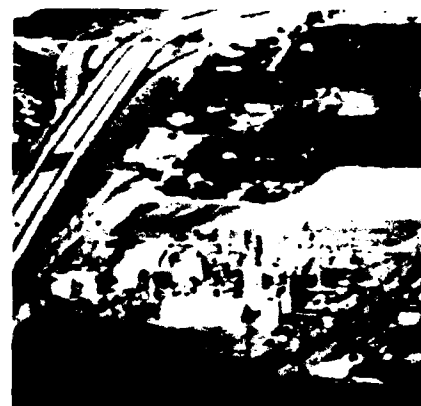
Frame No. 8



Frame No. 12



Frame No. 12



Frame No. 16

Some Frames of the Original Head and Shoulders (on the Left) and Chemical Plant (on the Right) Data Sets.

Figure 1-3

adaptive scheme based on the classification of motion. Results for both these schemes have been presented primarily for the purpose of comparison with the schemes of Chapter IV and V.

Chapter IV starts with the basic concepts of transform coding. Then the results of intraframe 2-D transform coding are presented. They form a good basis for comparing the interframe 3-D transform as well as hybrid coding against intraframe coding and provide a measure of compression gain due to interframe redundancy. Finally, results of some non-adaptive and adaptive interframe transform coding are presented.

Chapter V presents non-adaptive and many adaptive hybrid coding schemes and forms the major portion of the interframe coding schemes of this thesis. Adaptive schemes include classification based on activity, motion measurement and compensation, and temporal subsampling with interpolation along motion trajectory.

In chapter VI we present a new method of data compression for transmission over noisy channels. This consists of joint optimization of source and channel coding to reduce the overall MSE distortion in the signal due to quantization and channel noise. Rate distortion curves for coding of random variables, and one and two dimensional random processes are given together with experiments on actual image data.

Summary and conclusions of our investigation as well as future areas of research are reported in chapter VII. Appendix A discusses the problem of modeling 2-D image statistics together with some results. Appendix B gives the results of comparisons of various unitary discrete transforms used in data compression for some 2-D random fields.

CHAPTER II

MODELING, MEASUREMENT, AND ANALYSIS OF TEMPORAL CHARACTERISTICS

The temporal characteristics of a sequence of images differ considerably for various applications. The changes between two consecutive frames basically have two components, deterministic, and random. If a pixel or a group of pixels in a current frame has a correspondence with a pixel or a group of pixels in the preceding frame which can be characterized by a deterministic function, then that function is the deterministic component and the residual value of the pixel in the current frame after subtracting the deterministic component will be called the random component.

In motion images, some common types of deterministic components are, linear translation or rotation of objects against a fixed background in a scene, zooming and panning of the camera, linear and rotational motion of the camera, etc. In practice, the interframe motion is a combination of the above and various other motions which are not easy to characterize.

2.1 Motion Characteristics from Temporal Cross-Sections

In a laboratory the interframe motion can be perceived by viewing the images as a movie. We have considered an alternative way of presenting the data so that the motion can be inferred by looking at the images in a stationary mode. Since many image processing facilities (including ours) do not have the capability to display interframe digital data in real time, this method is useful in visual representation of interframe motion by stationary images.

We select a line in any direction, say θ , in the image plane passing through the region of interest and store the pixels along that line within the region of interest as a horizontal line of another image A_θ . Then we select lines from the successive frames at the same spatial location and store them one below another at vertical sampling intervals of the images. The resulting line sampled image A_θ over the time period of interest is a temporal cross-section in the direction θ . The inter-frame motion could then be visually analyzed by viewing several temporal cross-sections as follows.

If a pixel is undergoing linear translation along the direction θ its path will appear as a curve on the plane of A_θ . The slope of the curve (with respect to the vertical axis) gives the velocity of the pixel. The pixel intensities on this curve will be constant. In the context of wave propagation, fluid flow (or more generally for systems described by hyperbolic partial differential equations) etc., these curves are called the characteristics. If a region is undergoing linear translation perpendicular to θ , the image of that region will appear on A_θ with a scaling factor along the vertical axis of A_θ (the scaling factor depends upon the velocity of the region). If there is a camera zoom, we will see lines converging or diverging along the zoom axis. If an object in the region is rotating, we will see sinusoidal traces. Such temporal cross-sections have been used earlier for tomography x-ray images in which the object is rotated at uniform speed (by making an equivalence between time and angle) and they are called sinograms [63].

Figure 2-1 shows temporal cross-sections for the Head and Shoulders and the Chemical Plant images along some spatial directions. From images



Head & Shoulders Frame 8
(a)



Chemical Plant Frame 8
(c)



Temporal Cross-Sections of the
Head & Shoulders Images Along
(i) Row #183
(ii) Row #185
(iii) Column #127
(iv) Column #129
(v) Main Diagonal
(b)



Temporal Cross-Sections of the
Chemical Plant Images Along
(i) Row #170
(ii) Row #172
(iii) Column #127
(iv) Column #129
(v) Main Diagonal
(d)

Temporal Cross-Sections of 16 Frames of Head and Shoulders and Chemical Plant Images. White lines on images in (a) and (c) show the spatial locations of the cross-sections.

Figure 2-1

(a) and (b) we see down and up and right to left motion of the face in the 16 frames of the Head and Shoulders. From images (c) and (d) we see that for the Chemical Plant images the motion is almost purely vertically downward. First slow, then rapid, and just before the end it is almost stationary. We also notice that the slope of the motion trajectories in d(iii) increases as we move from left to right, which means that the bottom of the images is moving faster than the top. Since the ground objects are stationary, it is only possible if the camera is having motion other than linear translation parallel to the ground.

2.2 Interframe Motion Trajectory Estimation and Modeling:

Let us assume that each point of a continuous image is undergoing motion and appears at some location in an image at other time instant. Let $u(x,y,t)$, a zero mean random variable, denote the intensity of the (x,y) coordinate of an image at time t . Let each image be a sample of a 2-D homogeneous stationary random process whose covariance is given by

$$E[u(x',y',t)u(x+x',y+y',t)] = \sigma^2 R(|x|,|y|), \quad (2-1)$$

where $E[\cdot]$ denotes the expectation, $|\cdot|$ denotes the absolute value, and σ^2 is the variance of $u(x,y,t)$.

Let $(x+dx, y+dy, t+dt)$ be the new location of the point (x,y,t) . Then the trajectory of motion is given by

$$u(x,y,t) = u(x+dx, y+dy, t+dt) = \text{Constant}. \quad (2-2)$$

Let the observed value of $u(x,y,t)$ be given by

$$v(x,y,t) = u(x,y,t) + \eta(x,y,t) \quad (2-3)$$

when η is the observation noise which is assumed to be white and independent of u . Let η have zero mean and variance σ_η^2 .

Now let us assume that the motion trajectory is estimated piecewise, i.e., at discrete time instants. Let \bar{dx} and \bar{dy} be the estimates of dx and dy , respectively, and

$$\tilde{dx} = dx - \bar{dx}, \quad \tilde{dy} = dy - \bar{dy}$$

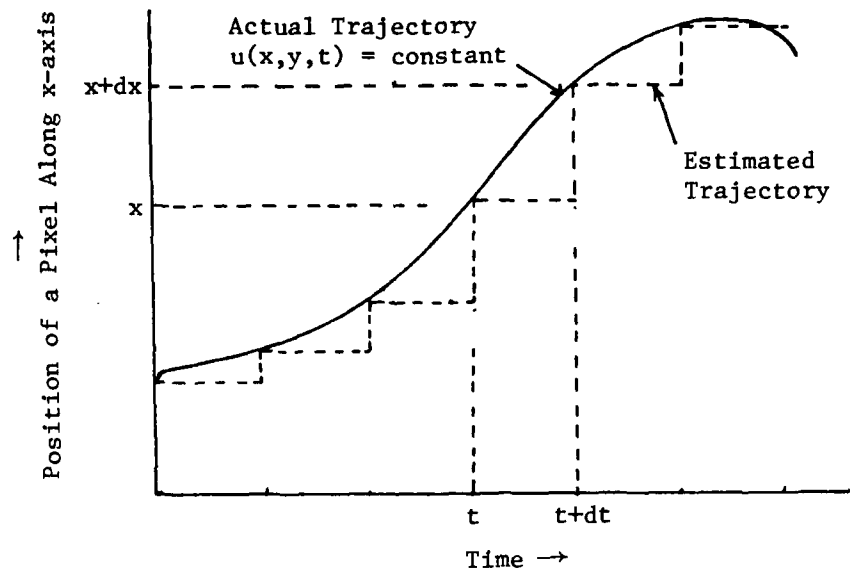
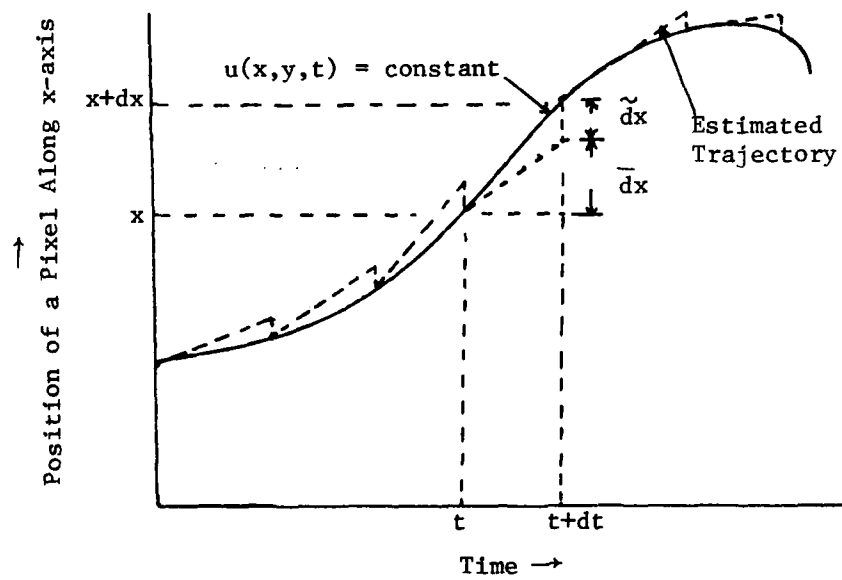
be the motion estimation error. Figure 2-2 shows the concept of trajectory approximation without and with motion estimation for the component of the motion along x-axis.

The motion compensated interframe estimate is given by

$$\bar{v}^c(x+\bar{dx}, y+\bar{dy}, t+dt) = v(x, y, t), \quad (2-4)$$

where superscript c denotes motion compensation. The temporal correlation after motion compensation is given by

$$\begin{aligned} \rho_{dt}^c &= \frac{E[\bar{v}^c(x+\bar{dx}, y+\bar{dy}, t+dt) \cdot v(x+\bar{dx}, y+\bar{dy}, t+dt)]}{E[v^2(x+\bar{dx}, y+\bar{dy}, t+dt)]} \\ &= \frac{E[v(x, y, t) \cdot v(x+\bar{dx}, y+\bar{dy}, t+dt)]}{E[\{u(x+\bar{dx}, y+\bar{dy}, t+dt) + \eta(x+\bar{dx}, y+\bar{dy}, t+dt)\}^2]} \\ &= \frac{1}{\sigma^2 + \sigma_\eta^2} \cdot E[\{u(x, y, t) + \eta(x, y, t)\} \cdot \{u(x+\bar{dx}, y+\bar{dy}, t+dt) + \eta(x+\bar{dx}, y+\bar{dy}, t+dt)\}] \\ &= \frac{1}{\sigma^2 + \sigma_\eta^2} \cdot E[u(x+\bar{dx}, y+\bar{dy}, t+dt) \cdot u(x+\bar{dx}, y+\bar{dy}, t+dt)] \\ &= \frac{1}{\sigma^2 + \sigma_\eta^2} \cdot E[\sigma^2 \cdot R(|dx - \bar{dx}|, |dy - \bar{dy}|)] \end{aligned}$$

(a) No Motion Compensation, $\bar{dx} = 0$ 

(b) Motion Compensation

Figure 2-2: Concept of Piecewise Trajectory Estimation

or

$$\rho_{dt}^c = \frac{\sigma^2 E[R(|\tilde{dx}|, |\tilde{dy}|)]}{\sigma^2 + \sigma_n^2}$$

or

$$\rho_{dt}^c = \frac{E[R(|\tilde{dx}|, |\tilde{dy}|)]}{1 + \sigma_n^2 / \sigma^2} \quad (2-5)$$

For small values of \tilde{dx} and \tilde{dy} most image covariance functions could be assumed to be approximately linear functions of $|\tilde{dx}|$ and $|\tilde{dy}|$ and the above could be approximated by

$$\rho_{dt}^c \approx \frac{R(E[|\tilde{dx}|], E[|\tilde{dy}|])}{1 + \sigma_n^2 / \sigma^2} \quad (2-6)$$

Thus, from the distribution of \tilde{dx} and \tilde{dy} one can obtain the temporal correlation, which may be used for interframe data compression.

We now define another quantity, which we call motion compensated interframe variance (MCIFV) as

$$\begin{aligned} \xi_{dt}^c &= E[\{v(x+dx, y+dy, t+dt) - \bar{v}^c(x+dx, y+dy, t+dt)\}^2] \\ &= E[\{v(x+dx, y+dy, t+dt)\}^2] + E[\{\bar{v}^c(x+dx, y+dy, t+dt)\}^2] \\ &\quad - 2E[v(x+dx, y+dy, t+dt) \cdot \bar{v}^c(x+dx, y+dy, t+dt)] \\ &= 2(\sigma^2 + \sigma_n^2) - 2\rho_{dt}^c(\sigma^2 + \sigma_n^2) \end{aligned}$$

$$\therefore \xi_{dt}^c = 2(\sigma^2 + \sigma_n^2)(1 - \rho_{dt}^c) \quad (2-7)$$

$$\text{or } \rho_{dt}^c = 1 - \xi_{dt}^c / 2(\sigma^2 + \sigma_n^2) \quad (2-8)$$

In the absence of motion compensation (2-5) and (2-7) become

$$\rho_{dt} = \frac{E[R(|dx|, |dy|)]}{1 + \sigma_n^2 / \sigma^2} \quad (2-9)$$

$$\xi_{dt} = 2(\sigma^2 + \sigma_{\eta}^2)(1 - \rho_{dt}) . \quad (2-10)$$

In the above discussion, the observation noise was included to show how it effects temporal correlation. In our coding experiments and future analysis, we assume that no observation noise is present.

2.3 Motion Measurement Techniques

In this section we describe methods of measuring interframe motion for digitized images with particular emphasis on data compression.

First, we approximate the interframe motion by piecewise linear translation of one or more areas of a frame relative to a reference frame. The segmentation of an image into areas each of which is undergoing approximately the same linear translation and the measurement of the magnitude and the direction of the linear translation of each area, is a difficult task. Cafforio and Rocca [11] describe a method for segmentation and measurement of the linear displacement of a single moving object in a stationary background. Then extension of the method to more than one moving object has also been shown. Thus, method becomes increasingly complex as the number of moving areas increases and the size of the image grow larger. There is another difficulty with such a method of segmentation if the information of segmentation and linear translation is to be coded. Coding of segmentation with arbitrary boundaries would require a complex scheme and, moreover, the length of the code will be large.

A simpler method is to segment an image into fixed size smaller rectangular areas and to assume that each of these areas is undergoing independent linear translation. If these areas are small enough, rotation, zooming, etc. of larger objects can be closely approximated by piecewise

linear translation of these smaller areas. Also, it avoids the problem of coding the segmentation information. Only the displacement vector of each of the areas need to be transmitted. Another simplification is to restrict the motion measurement to an integer number of pixels. This would give an accuracy up to .5 pixels in the moving areas. Since in practice the motion is not an ideal linear translation, an effort to estimate the displacement vector up to a fraction of a pixel will not result in significant improvement in prediction.

A method which has been used for the measurement of linear shift between two given images, particularly for aerial guidance, is area correlation [59,85]. This consists of calculating the area correlation function of the two images. The location of the peak of the correlation function gives the displacement vector. The area correlation function is usually calculated via the fast Fourier transform (FFT). To improve the accuracy of this method some filtering or preprocessing of the images is required, which could be done in the spatial domain [59] or the Fourier domain [85].

For the purpose of piecewise linear translation measurement we divide an image into smaller rectangular areas, which we call sub-blocks, and correlate them with the appropriate areas of the reference image. Let U be an $M \times N$ size sub-block of an image and U_R be an $(M+2p) \times (N+2p)$ sub-block of the reference image, centered at the same spatial location as U , where p is the maximum displacement allowed in integer number of pixels in either direction. Then the area correlation function is given by

$$C^f(i,j) = \sum_{m=1}^M \sum_{n=1}^N u^f(m,n) u_R^f(m+i,n+j), \quad -p \leq i,j \leq p \quad (2-11)$$

where superscript f denotes filtering or preprocessing in the spatial or Fourier domain. A simple spatial operator which has been found to be useful in area correlation is a four point Laplacian operator given by

$$u^f(m,n) = u(m,n) - \{u(m-1,n) + u(m+1,n) + u(m,n-1) + u(m,n+1)\}/4, \quad \forall m,n.$$

Let V and V^f be the discrete Fourier transforms of U and U^f , respectively, then a Fourier domain filter given by

$$|v^f(m,n)| = |v(m,n)|^\gamma, \quad 0 \leq \gamma \leq 1 \quad (2-12)$$

where $|\cdot|$ represents the magnitude, has been found to be useful [85].

We have found that the performance of the area correlation method is poor for smaller sub-block sizes, areas of low spatial activity, and for sub-blocks not undergoing pure linear translation. We have found another method which does significantly better under most circumstances for interframe image motion estimation. This method requires a search for the direction of minimum distortion (or DMD) and is described below.

Let us define a mean distortion function between U and U_R as

$$D(i,j) = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N g\{u(m,n) - u_R(m+i,n+j)\}, \quad -p \leq i,j \leq p \quad (2-13)$$

where $g\{x\}$ is a given positive and increasing distortion function of x , e.g., $g\{x\} = x^2$ would correspond to $D(i,j)$ as mean square error function. The direction of minimum distortion is given by (i,j) , such that $D(i,j)$ is minimum.

One difficulty with finding DMD as stated above is that it requires evaluation of $D(i,j)$ for $(2p+1) \times (2p+1)$ directions, and even for motions

up to 5 pixels along either side of the axes, one has to search 121 directions. We have found a solution to overcome the above difficulty by making an assumption that if

$$D_0(q, \ell) = \min_{i, j} \{D(i, j)\}$$

then for $m = i - q$, $n = j - \ell$, the functions

$$D_1(|m|, |n|) = D(i, j) - D_0(q, \ell), \quad m \geq 0, n \geq 0$$

$$D_2(|m|, |n|) = D(i, j) - D_0(q, \ell), \quad m \leq 0, n \leq 0$$

$$D_3(|m|, |n|) = D(i, j) - D_0(q, \ell), \quad m \leq 0, n \geq 0$$

$$D_4(|m|, |n|) = D(i, j) - D_0(q, \ell), \quad m \geq 0, n \leq 0$$

are nondecreasing function of $|m|$ and $|n|$, i.e., for $m, n, m', n' \geq 0$,
 $1 \leq k \leq 4$,

$$D_k(m, n) \leq D_k(m', n'), \quad \text{if } m \leq m' \text{ and } n \leq n'.$$

For $g\{x\} = x^2$ the above is equivalent to the assumption that the covariance function of images is a nonincreasing function. Most image covariance functions satisfy this condition, at least in a small neighborhood of the origin.

Having made the above assumption, we use a 2-D directed search method, which is similar to the binary or logarithm search [90] in one dimension. The search is accomplished by successively reducing the area of search to half or less. Each step consists of searching five directions, which contain the center of the area, and the midpoints between the center

and the four boundaries of the area along the x,y axes passing through the center. This procedure continues until the plane of search reduces to a 3×3 size. In the final step all the 9 directions are searched and the location corresponding to the minima is the DMD. The algorithm is given below.

For any integer $m > 0$, we define

$$\mathcal{N}(m) = \{(i,j) ; -m \leq i,j \leq m\}$$

$$\mathcal{M}(m) = \{(0,0), (m,0), (0,m), (-m,0), (0,-m)\}.$$

A 2-D Logarithmic Search Procedure for DMD:

Step 1: (initialization)

$$D(i,j) = \infty \quad (i,j) \notin \mathcal{N}(p)$$

$$n' = \lfloor \log_2 p \rfloor$$

$$n = \max\{2, 2^{n'-1}\}$$

$$q = l = 0 \quad (\text{or an initial guess for DMD})$$

where $\lfloor \cdot \rfloor$ is a lower integer truncation function.

Step 2: $\mathcal{M}'(n) \leftarrow \mathcal{M}(n)$.

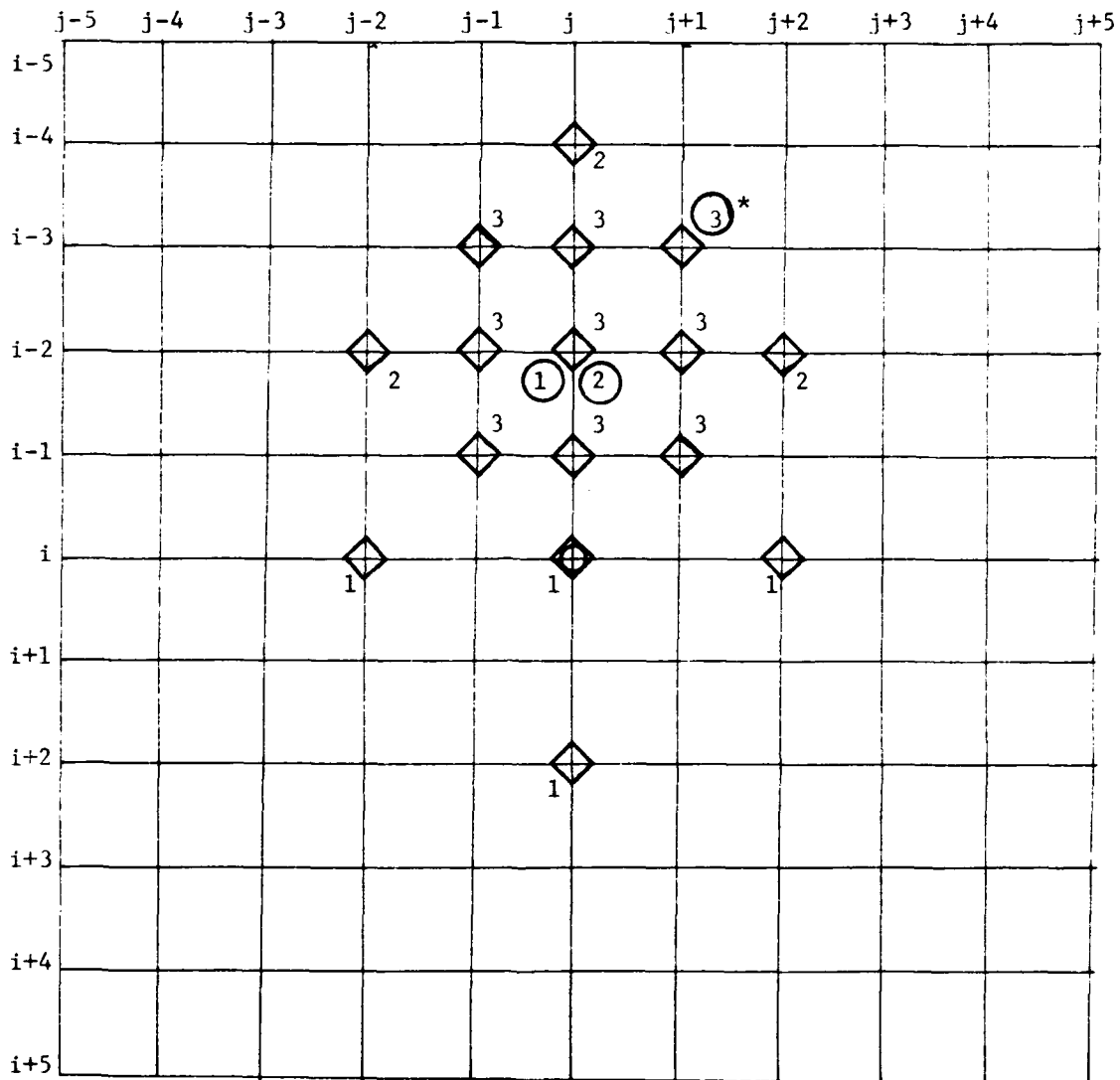
Step 3: Find $(i,j) \in \mathcal{M}'(n)$ such that $D(i+q, j+l)$ is minimum. If $i=0$ and $j=0$, go to Step 5; otherwise go to Step 4.

Step 4: $q \leftarrow q+i, l \leftarrow l+j; \mathcal{M}'(n) \leftarrow \mathcal{M}'(n) \cap (-i, -j)$; go to Step 3.

Step 5: $n \leftarrow n/2$. If $n=1$, go to Step 6; otherwise, go to Step 2.

Step 6: Find $(i,j) \in \mathcal{M}'(1)$ such that $D(i+q, j+l)$ is minimum. $q \leftarrow i+q, l \leftarrow l+j$. (q,l) then gives the DMD.

Figure 2-3 illustrates the search procedure for $p = 5$.



The figure above shows the concept of the 2-D logarithmic search to find a pixel in another frame which is registered with respect to the pixel (i,j) of a given frame, such that the mean-square error over a block defined around (i,j) is minimized. The search is done step by step with \diamond indicating the directions searched at a step number marked. The numbers circled show the optimum directions for that search step and the * shows the final optimum direction, (i-3, j+1) in the above example. This procedure requires only searching 13-21 directions for the above grid as opposed to 121 total possibilities.

Figure 2-3: A 2-D Logarithmic Search Procedure for the Direction of Minimum Distortion.

2.4 Motion Measurement Results

The method of DMD motion measurement, discussed in the previous section, was applied to the Head and Shoulders and the Chemical Plant data. The distortion function $g(x) \approx x^2$ was used so that DMD would correspond to minimum mean square error in registration of the sub-blocks. A sub-block size of 16×16 was chosen. The sizes 16×16 and 32×32 were found to be good compromises between accuracy of piecewise linear translational approximation of the motion, the cost of transmitting displacement vectors, and the complexity of data compression schemes using motion measurement.

For the multiframe data, when the reference image is a neighboring frame of the image relative to which motion measurement is done, the quantity $D(0,0)$ will be called interframe variance. Once the DMD for a sub-block has been found, the area of the reference image in the direction of DMD is taken as the motion compensated estimate of the sub-block. By collecting all the motion compensated estimates from a reference frame, one obtains the motion compensated reference frame. If (q,l) is the displacement vector of the DMD, then $D(q,l)$ is the interframe variance with motion compensation and $D(0,0)$ is the interframe variance without motion compensation for that sub-block. These quantities for a frame are obtained by averaging them over all the sub-blocks.

When the reference frame is the same as the current image itself, $D(i,j)$ computed over complete frame gives the average interframe variance as a function of uniform linear translation vector (i,j) . The above quantity gives a rough estimate of the average motion between two frames. Figure 2-4 gives the interframe variance as a function of linear translation for the Head and Shoulders and the Chemical Plant data.

$j \rightarrow$

	0	1	2	3	4	5	6	7	8	9
0	0.	201.24	590.15	906.41	1114.52	1252.31	1352.64	1435.52	1505.76	1604.23
1	395.51	572.96	850.12	1081.50	1232.57	1334.16	1414.27	1485.62	1544.27	1640.74
2	656.41	1045.02	1246.83	1377.66	1456.26	1518.01	1572.85	1620.97	1666.99	1757.12
3	1203.14	1406.69	1522.43	1601.17	1653.98	1697.36	1736.36	1775.95	1816.04	1907.45
4	1557.26	1656.61	1738.08	1786.67	1821.18	1851.61	1879.28	1913.34	1960.95	2056.41
5	1777.79	1838.89	1892.33	1929.44	1954.79	1976.33	2003.06	2042.76	2095.15	2192.30
6	1949.76	1995.31	2033.46	2054.44	2068.68	2085.93	2113.29	2158.59	2217.21	2314.32
7	2106.37	2129.62	2150.60	2164.52	2177.94	2194.67	2234.69	2281.92	2334.45	2422.27
8	2259.66	2251.57	2261.73	2265.31	2275.10	2300.32	2343.06	2396.38	2451.38	2530.20
9	2356.43	2357.13	2360.90	2367.97	2384.15	2413.54	2456.44	2506.30	2551.13	2622.64

$i \downarrow$

(a) Chemical Plant Frame No. 12

$j \rightarrow$

	0	1	2	3	4	5	6	7	8	9
0	0.	74.69	260.54	493.29	732.43	960.65	1172.63	1370.55	1556.49	1746.05
1	65.53	173.45	360.80	589.92	824.99	1049.99	1259.65	1454.63	1636.09	1824.73
2	250.93	345.22	525.49	739.69	964.13	1180.75	1383.59	1573.07	1751.55	1933.13
3	462.50	559.45	726.29	927.53	1137.78	1342.53	1535.58	1716.93	1888.14	2063.54
4	693.52	791.32	948.47	1136.37	1332.77	1524.48	1706.13	1877.65	2040.32	2206.22
5	926.79	1026.79	1176.81	1354.25	1538.60	1717.94	1887.63	2048.75	2202.17	2361.92
6	1160.54	1256.68	1403.60	1572.92	1747.23	1915.59	2074.43	2224.69	2368.91	2520.25
7	1385.66	1483.01	1624.69	1787.37	1953.25	2112.06	2266.94	2401.66	2530.65	2679.79
8	1603.09	1700.73	1837.66	1994.61	2153.11	2303.61	2443.58	2575.30	2701.66	2837.57
9	1811.46	1907.91	2041.60	2192.09	2344.07	2487.32	2619.38	2742.94	2861.79	2990.40

$i \downarrow$

(b) Head & Shoulders Frame No. 8

Figure 2-4: Interframe Variance as a Function of Linear Translation
(i,j) Measured over One Complete Frame

Figures 2-5 and 2-6 show the results of motion measurement for the central 256×128 portion of a frame for the Head and Shoulders and the Chemical Plant data, respectively, relative to the preceding frame. Part (a) shows the displacement vectors for each of the 16×16 sub-blocks. The success of the DMD motion location method seems evident from Figure 2-6(a). The image is known to have a vertically downward motion relative to the previous frame as well as a geometric distortion such that the top of the image undergoes a smaller displacement and the bottom a larger displacement [67, p. 102]. We can see that mostly our scheme predicts a successively larger motion as one moves from the top to the bottom, as expected.

Parts (b) and (c) of Figure 2-5 and 2-6 show the interframe variance (IFV) without and with motion compensation (MC) for various sub-blocks. For the Head and Shoulders data (Figure 2-5) there are very wide variations among sub-blocks in IFV without motion compensation. This is both due to a variation in motion (from nearly stationary to more than 3-4 pixels/frame) and the spatial activity (from a pixel to pixel correlation of .99 to less than .8) as a result of non-stationarity. After motion compensation the variation narrows down significantly and is mostly due to the spatial activity. For Chemical Plant (Figure 2-6) the variation without motion compensation is not as wide because there are no stationary areas.

Comparing the average values of the IFV with motion compensation and the entries of Figure 2-4, we can compute an estimate of the average motion uncertainty. We assume that the motion uncertainty is identically distributed along both the image dimensions and that, for small motion, the IFV is a linear function of motion in pixels. With this assumption

(0, 0)	(0, 0)	(0, 0)	(0, 1)	(0, 0)	(0, 0)	(0, 1)	(0, 1)
(0, 0)	(-2, 0)	(-3, -1)	(0, 1)	(1, 0)	(0, 1)	(0, 2)	(0, 1)
(4, 0)	(-1, 1)	(-2, -2)	(0, 2)	(0, 2)	(0, 2)	(0, 2)	(0, 2)
(1, 0)	(4, 0)	(0, 1)	(0, 1)	(0, 2)	(0, 2)	(0, 2)	(1, 2)
(0, 0)	(-2, 0)	(1, 1)	(-1, 2)	(-1, 2)	(-1, 2)	(0, 2)	(0, 2)
(-1, 0)	(-2, 0)	(-1, 1)	(-1, 2)	(-1, 2)	(-1, 2)	(-1, 2)	(-1, 2)
(5, 0)	(1, 1)	(-2, 2)	(-1, 2)	(-1, 2)	(-1, 2)	(-1, 2)	(-1, 2)
(-1, 0)	(-4, 1)	(-2, 1)	(-2, 2)	(-1, 2)	(-1, 3)	(-1, 2)	(-1, 2)
(5, 0)	(-2, 1)	(-1, 1)	(-2, 2)	(0, 3)	(-1, 2)	(-2, 2)	(-1, 1)
(-1, 0)	(-4, 1)	(-2, 1)	(-2, 2)	(-2, 2)	(-2, 2)	(-2, 2)	(-2, 1)
(-5, 0)	(1, 0)	(-2, 1)	(-3, 1)	(-3, 1)	(-2, 2)	(-3, 2)	(-2, 1)
(-1, 0)	(-2, -2)	(-2, -1)	(-3, 1)	(-3, 1)	(-3, 2)	(-2, 1)	(-1, 0)
(-2, -1)	(-2, -3)	(-2, -1)	(-3, 1)	(-3, 1)	(-3, 1)	(-2, 0)	(-2, 0)
(-4, 1)	(-4, 2)	(-2, 0)	(-1, 0)	(-3, 1)	(-3, 0)	(-3, 1)	(-2, 0)
(-4, 1)	(-4, 0)	(-4, 1)	(-2, 0)	(-3, 0)	(-3, 0)	(-3, 0)	(-2, 0)
(-3, 1)	(-3, 1)	(-4, 0)	(-2, 0)	(-4, 1)	(-4, 0)	(-3, 0)	(-4, 0)

(a) Coordinates of the Displacement Vector, (q, ℓ) , for Sub-blocks

9.05	11.49	11.95	12.06	9.05	6.59	4.40	5.21
13.66	11.27	12.53	17.82	45.30	20.90	46.47	6.33
16.79	9.86	11.43	191.25	19.96	13.65	23.80	21.17
7.36	9.37	29.54	80.03	48.90	84.77	29.30	80.20
7.21	11.50	27.77	38.26	53.89	75.72	41.20	69.59
28.78	13.01	63.32	40.93	51.97	99.33	53.37	66.25
26.65	11.55	362.73	636.59	332.21	197.31	80.84	52.75
12.23	25.26	284.87	245.84	647.09	825.64	160.60	107.85
11.89	15.61	48.60	419.11	721.41	99.33	78.15	43.96
10.46	13.59	75.77	430.92	358.83	65.70	69.69	34.75
19.75	7.95	61.29	667.21	1410.71	368.22	24.56	7.84
27.43	637.86	994.14	1298.97	669.24	135.75	40.05	128.11
824.11	246.85	1.86	1406.02	1614.05	68.70	368.27	722.14
1.66	1.97	1.92	37.86	2805.94	1866.39	990.37	667.69
1.69	2.42	1.54	214.66	2424.13	912.05	2072.52	57.64
1.73	1.74	2.07	451.98	780.09	206.96	1555.71	1.33

(b) IFV Without MC, $D(0,0)$, for Sub-blocks. Average for Frame = 156.70

9.05	11.49	11.95	9.90	9.05	6.59	4.00	5.12
13.68	9.53	10.34	17.10	41.24	19.96	53.44	3.86
11.91	6.03	10.12	22.85	3.23	2.51	3.90	6.95
7.33	8.16	12.84	16.65	13.51	12.42	10.63	15.65
7.21	10.41	13.72	14.60	14.07	14.70	11.90	14.05
17.46	10.50	19.35	17.25	12.75	13.56	10.64	6.85
15.46	9.77	32.21	25.00	21.61	16.54	11.82	13.55
11.48	20.29	45.25	21.02	56.34	59.11	11.35	10.03
7.86	13.41	14.52	18.97	45.00	15.33	16.85	9.94
9.47	9.52	26.52	32.91	15.46	14.27	6.60	9.86
16.00	7.55	18.50	77.89	116.35	22.21	6.75	5.70
12.07	10.71	21.55	71.91	23.62	21.29	8.57	15.02
11.16	9.51	1.19	40.34	9.32	6.24	15.07	53.92
1.03	.87	1.04	7.90	53.63	40.15	17.54	17.67
1.10	1.33	.84	56.75	108.65	58.47	85.49	3.16
1.54	1.17	.86	54.62	117.63	141.74	28.55	.94

(c) IFV with MC, $D_0(q, \ell)$, for Sub-blocks. Average for Frame = 15.24Figure 2-5: Results of Motion Measurement Relative to the Previous Frame on a Portion of Head & Shoulders Frame No. 8 for Sub-block Size of 16×16 .

(-2, 0) (-2, 0) (-2, -1) (-2, -1) (-2, -1) (-2, 0) (-2, 0) (-2, 0)
 (-2, 0) (-2, 0) (-2, 0) (-2, 0) (-2, 1) (-2, 0) (-2, 0) (-2, 0)
 (0, -1) (-2, 0) (-2, 0) (-2, 0) (-2, 0) (-2, 0) (-4, 1) (-4, 1)
 (0, -1) (-2, 0) (-2, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0)
 (-2, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 1) (-3, 0) (-2, 0)
 (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-2, 0)
 (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-4, 0)
 (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0) (-3, 0)
 (-3, 0) (-3, 1) (-3, 0) (-3, 0) (-4, 0) (-3, -1) (-3, 0) (-2, 1)
 (-3, 0) (-4, 0) (-3, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)
 (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-3, 0)
 (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)
 (-4, 0) (-4, 0) (-4, -4) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)
 (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)
 (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)
 (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0) (-4, 0)

(a) Coordinates of the Displacement Vector, (q, ℓ) , for Sub-blocks

511.64	597.27	571.14	632.71	827.03	922.35	645.44	1459.20
1057.11	1089.21	1124.89	1050.01	1693.84	1021.40	1243.70	229.61
784.87	520.21	231.55	671.46	1160.05	505.98	462.54	327.66
686.68	99.36	255.40	661.92	1127.94	1307.86	649.19	1522.20
219.39	825.07	2028.18	1108.00	1367.15	713.44	327.52	550.58
428.77	443.60	993.16	1460.37	1540.19	742.23	716.89	148.07
547.96	1977.13	2961.50	1103.89	1290.16	1526.96	1227.73	1153.74
568.26	1169.63	1174.92	1279.52	2084.34	1837.81	557.45	441.46
853.45	587.57	386.27	670.85	423.16	224.12	315.39	365.22
1604.13	1842.45	1534.98	864.91	856.37	1014.78	1112.66	1440.28
1025.23	1911.53	1128.82	1152.35	957.78	2256.59	2447.02	2715.26
916.19	2240.13	4895.79	3959.52	4558.25	3279.82	2575.70	3118.23
324.22	1546.33	2458.47	2051.43	1462.06	1313.68	2093.52	2658.85
476.14	99.41	271.03	698.52	662.27	1305.63	3714.74	2868.21
58.86	257.09	136.53	146.07	1049.13	859.63	713.23	2783.77
57.52	111.50	97.30	881.49	1194.90	543.20	253.80	418.03

(b) IFV without MC, $D(0,0)$, for Sub-blocks. Average for the Frame = 1327.64

125.96	104.29	215.44	209.06	291.97	539.61	264.70	447.76
112.47	212.06	151.51	134.60	120.76	121.80	146.09	54.83
145.24	128.81	75.67	125.20	164.47	113.45	81.66	64.29
169.34	69.10	64.04	65.56	94.01	78.70	156.57	147.68
75.16	119.13	174.16	154.67	163.51	124.25	93.13	150.46
99.97	111.45	129.74	121.82	154.14	63.16	125.89	75.50
104.53	161.10	261.61	134.12	113.86	134.53	160.80	176.09
156.12	149.27	100.63	108.84	166.15	193.95	211.07	361.14
161.34	80.67	65.43	104.45	142.71	104.00	189.84	283.55
198.67	300.35	258.99	220.72	200.45	164.16	256.12	343.75
194.85	289.87	166.10	150.98	205.83	259.12	390.72	440.61
184.13	167.12	204.16	250.51	262.75	264.55	474.81	452.71
69.38	121.95	131.85	203.39	126.57	234.56	221.32	255.22
29.79	19.98	36.44	68.70	116.20	132.47	164.75	148.71
20.07	21.48	18.81	14.70	41.66	124.66	126.66	112.06
4.90	11.63	12.72	26.44	123.60	19.24	48.48	40.98

(c) IFV with MC, $D_0(q, \ell)$, for Sub-blocks. Average for the Frame = 139.34

Figure 2-6: Results of Motion Measurement Relative to the Previous Frame on a Portion of Chemical Plant Frame No. 12 for Sub-block Size of 16×16 .

we compare the IFV with the diagonal entries of Figure 2-4 and find the average motion by interpolation. For example, the average IFV with motion compensation for the Chemical Plant frame No. 12 is 139.34 (Figure 2-6(c)). Comparing this value with the diagonal entries of Figure 2-4(a) we find that it lies between 0 and 572.96. Thus, by interpolation we obtain an average value of motion uncertainty after motion compensation as (.25,.25). Similarly, for the Head and Shoulders data we find it to be approximately (.1,.1). Observing that for the Head and Shoulders data more than half the image area is nearly stationary, the average motion uncertainty in the moving areas can be approximated as (.25,.25) pixel. This means that the DMD method indeed measures the motion with an accuracy up to the nearest integer pixel most of the time. Thus, the absolute value of the motion uncertainty in the moving areas can be modeled as uniformly distributed between 0 and .5 pixel along each of the dimensions, giving an average value of .25.

Table 2-1 shows the improvement in the interframe prediction due to motion compensation. With no motion compensation, the prediction of a pixel is the value of a pixel in the previous frame, having the same spatial location whereas with motion compensation, the predicted value comes from the previous frame pixel in the direction of minimum distortion. Note that the variance of this error is nothing but IFV. An interesting and important observation is that there is a wide frame to frame variation in IFV without motion compensation due to variation in motion activity as a function of time (4.64 dB between Head and Shoulders frames 6 and 7). After motion compensation this variation becomes negligible. Showing that the uncertainty in motion prediction is identically distributed over

TABLE 2-1

IMPROVEMENT IN SNR OF INTERFRAME PREDICTION ERROR (IFPE) DUE TO MOTION COMPENSATION. MOTION MEASUREMENT WAS DONE ON 16×16 SIZE SUB-BLOCKS USING DIRECTION OF MINIMUM DISTORTION SEARCH WITH MEAN SQUARE CRITERION.

DATA SET	FRAME No.	SNR OF IFPE IN DECIBELS		
		Without Motion Compensation	With Motion Compensation	Improvement
HEAD & SHOULDERS	6	29.90	35.88	5.98
	7	25.26	35.68	10.42
	8	26.18	36.30	10.12
	9	26.03	36.26	10.23
CHEMICAL PLANT	11	16.66	26.77	10.11
	12	16.90	26.69	9.79
	13	17.53	26.56	9.03

TABLE 2-2

IMPROVEMENT IN SNR OF INTERPOLATED FRAME (FROM THE PRECEDING AND THE FOLLOWING FRAMES OF THE ORIGINAL DATA) DUE TO MOTION COMPENSATION.

INTERPOLATED FRAME	SNR OF INTERPOLATED FRAME IN DECIBELS		
	Without Motion Compensation	With Motion Compensation	Improvement
HEAD & SHOULDERS FRAME #8	30.48	38.61	8.15
CHEMICAL PLANT FRAME # 12	19.34	29.56	10.22

different frames even though the motion itself is not. Figures 2-7(a) and 2-8(a) show interframe prediction and error images.

The error images shown in Figures 2-7 and 2-8 and elsewhere show the absolute value of the errors amplified and truncated to the largest value of 255. The darker points show larger errors. The amplification for the Head and Shoulders data is ten times and for the Chemical Plant data it is five times. Only about three-fourths portion of the error images have been shown for these data sets.

2.5 Frame Repetition and Interpolation Along Motion Trajectory

Frame skipping is one of the simplest methods of data compression for interframe motion images. For simplicity of discussion, we assume skipping of every alternate frame. However, the discussion could be easily extended to other cases. A skipped frame is generally reproduced by either repeating the preceding frame or by interpolation from the preceding and the following frames. Both these methods have serious effects on the quality of motion reproduction. The former results in jerkiness in the reproduction of the motion and the latter in blurring of the moving areas. This is evident by looking at part (i) of Figures 2-7(b) and 2-8(b).

Let U_{2k} be a sub-block of the $(2k)$ th frame where frames $2, 4, \dots, 2k, \dots$ have been skipped. Then U_{2k}^* , the reproduced value of U_{2k} , is obtained (without motion compensation) as follows.

Frame Repetition:

$$u_{2k}^*(m, n) = u_{2k-1}(m, n) \quad (2-14)$$



(i) Along Temporal Axis, SNR = 26.17 dB



(ii) Along Motion Trajectory, SNR = 36.31 dB



(a) Frame Repetition (or Interframe Prediction) Based on the Preceding Frame



(i) Along Temporal Axis, SNR = 30.47 dB



(ii) Along Motion Trajectory, SNR = 38.62 dB



(b) Frame Interpolation from the Preceding and the Following Frames

Effects of Motion Compensation on Interframe Prediction and Interpolation of Head and Shoulders Frame 8.

Figure 2-7



(i) Along Temporal Axis, SNR = 16.90 dB



(ii) Along Motion Trajectory, SNR = 26.69 dB



(a) Frame Repetition (or Interframe Prediction) Based on the Preceding Frame



(i) Along Temporal Axis, SNR = 19.34 dB

(ii) Along Motion Trajectory, SNR = 29.56 dB

(b) Frame Interpolation from the Preceding and the Following Frames

Effects of Motion Compensation on Interframe Prediction and Interpolation of Chemical Plant Frame 12

Figure 2-8

Frame Interpolation:

$$u_{2k}^*(m,n) = \frac{1}{2}\{u_{2k-1}(m,n) + u_{2k+1}(m,n)\}. \quad (2-15)$$

The disadvantages of simple frame repetition or interpolation can be overcome by using motion compensation, i.e., making the prediction or interpolation along the motion trajectory. Using motion compensation (2-14) and (2-15) are replaced by

$$u_{2k}^*(m,n) = u_{2k-1}(m+q,n+l) \quad (2-16)$$

and

$$u_{2k}^*(m,n) = \frac{1}{2}\{u_{2k-1}(m+q,n+l) + u_{2k+1}(m+q',n+l')\} \quad (2-17)$$

respectively, where (q,l) and (q',l') are the coordinates of the displacement vectors of U_{2k} relative to the preceding and the following frames, respectively.

The improvement in SNR of the interframe prediction error shown in Table 2-1 and Figures 2-7(a) and 2-8(a) is nothing but the improvement due to frame repetition along motion trajectory compared with a simple frame repetition. Table 2-2 and Figures 2-7(b) and 2-8(c) show the improvement due to the frame interpolation along the motion trajectory compared with a simple interpolation along the temporal axis.

Thus, we see that the approximation of the motion by linear translation, on a sub-block by sub-block basis, gives excellent results for the video motion images considered. These results could be used with interframe predictive coding schemes, such as DPCM, and still better with hybrid coding schemes (as discussed in chapter V) with a great improvement in coder performance. The results of chapter V show an

improvement in compression gain by a factor of two when motion compensation and frame interpolation along motion trajectory are used. Even higher gains are expected with further increase in the sampling interval along the temporal axis (i.e., skipping more frames) and interpolation of missing frames along motion trajectory.

CHAPTER III

INTERFRAME PREDICTIVE CODING

In section 1.3.1 we had briefly discussed several predictive techniques. One of these techniques, called frame replenishment with cluster coding, is described in the following section. In section 3.2 we report a simple predictive scheme which we call adaptive classification prediction. This has been developed primarily for comparison with other schemes. These schemes have been simulated and applied on the Head and Shoulders images.

3.1 Frame Replenishment with Cluster Coding

This technique was developed at the Bell Telephone Laboratories and is described by Candy, et al. [13]. We have implemented it for comparison purposes. This technique mainly consists of transmitting the addresses and quantized amplitudes of the "significant" interframe differences of the consecutive frames. The interframe difference at any pixel location is classified as significant when its absolute value exceeds a fixed threshold. The experimental observation that most of the significant interframe differences occur in clusters along any frame line, motivates the fact that the addresses of the significant interframe differences could be efficiently coded by transmitting the beginning address of a cluster and a cluster terminator code. It is obvious that most of the clusters will appear in the areas of an image consisting of moving edges or objects. This implies that the technique would generate a variable bit-rate for each frame, depending upon the activity and motion in the frame. Thus, transmission of the data on a channel with a fixed bit-rate would necessitate a buffer.

To avoid an arbitrarily large buffer requirement, some controls (not without the penalty of higher distortion), which limit the buffer requirement to a given buffer length, are applied. In our simulations we keep the buffer capacity equal to the average number of coded bits (desired bit rate) per frame. All controls are determined by the number of bits residing in the buffer. Figure 3-1 describes the buffer control levels for this scheme.

If the contents of the buffer fall below point A, the next line is transmitted as it is, using 8 bits/pixel, to prevent buffer under-flow. When the buffer contents exceed point C, the frame differences in a cluster are subsampled, i.e., every other frame difference is transmitted and at the receiver the missing value is interpolated. The sub-sampling continues until the buffer contents fall below point B. When the buffer contents exceed points C, D and E, the threshold for classification of significant changes is increased successively to lower the number of significant changes. When the buffer is filled beyond point F, coding is stopped for one frame period and sub-sampling is continued for the next frame period. This is done to avoid the buffer overflow.

In the beginning, the first three lines of the first frame are force updated, i.e., they are transmitted as 8-bits/pixel. In the next frame, the next three lines are force updated and so on, except when the coder is in buffer overflow condition. At this rate a complete frame is refreshed or updated approximately every 3.5 seconds for 24 frames/sec. transmission. Nine-bit codes are used to designate starting of a new frame, starting of a new line, and the starting addresses of the clusters in a given line. Removal of isolated points of significant changes,

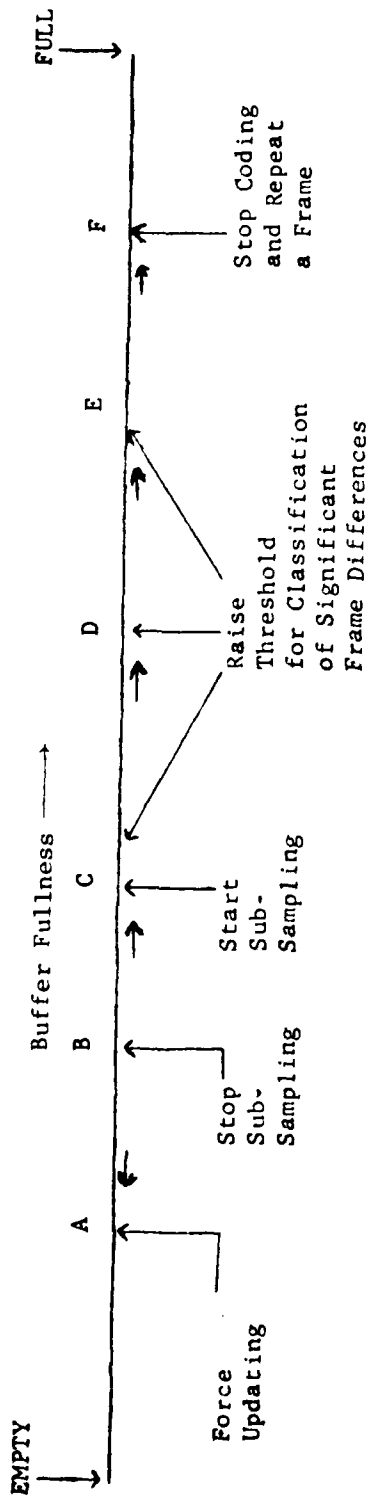


Figure 3-1: Buffer Control Levels for Frame Replenishment Cluster Coding

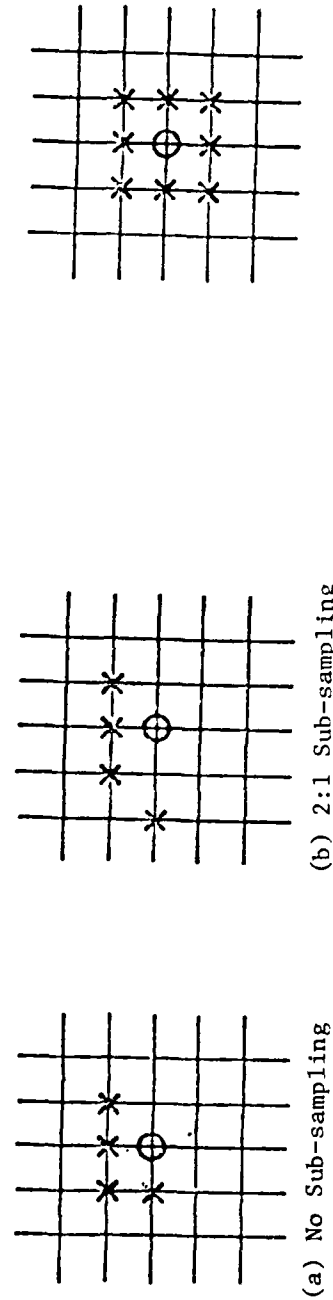


Figure 3-3: Nearest Neighbors in the Previous Frame for Adaptive Classification Prediction Scheme

Figure 3-2: Nearest Neighbors in the Present Frame for Adaptive Classification Prediction Scheme

and bridging of the clusters which are very close are done to reduce the bit-rate. For further details, see [13].

3.2 Adaptive Classification Prediction Coding

As pointed out in the last chapter, if a point moves more than one and if its direction of motion is unknown, then (for highly correlated images) spatial prediction performs better than pure temporal prediction. On the other hand, for stationary pixels a temporal prediction is preferable.

We have developed a very simple criterion to classify a pixel as stationary or slowly moving (about 1 pixel/frame in any direction) or rapidly moving relative to the previous frame. This could be easily implemented online and is based on the interframe differences of its nearest neighbors in the present frame as shown in Figure 3-2. Only in the case of slow motion do we approximate any kind of motion or change by a translatory motion, and we search its direction assuming that it came from one of the nearest neighbors in the previous frame as shown in Figure 3-3. The direction of the rapid motion is of no consequence to us as we rely on spatial prediction in this case. A block diagram of the scheme is shown in Figure 3-4.

Let $u_{k,i,j}$ denote the intensity of the j^{th} pixel on the i^{th} scan line of the k^{th} frame and $u_{k,i,j}^*$ be its reconstructed value at the receiver with no channel errors.

3.2.1 Motion Predictor - As established in the last chapter, the motion of a pixel is quite close to that of its nearest neighbors and

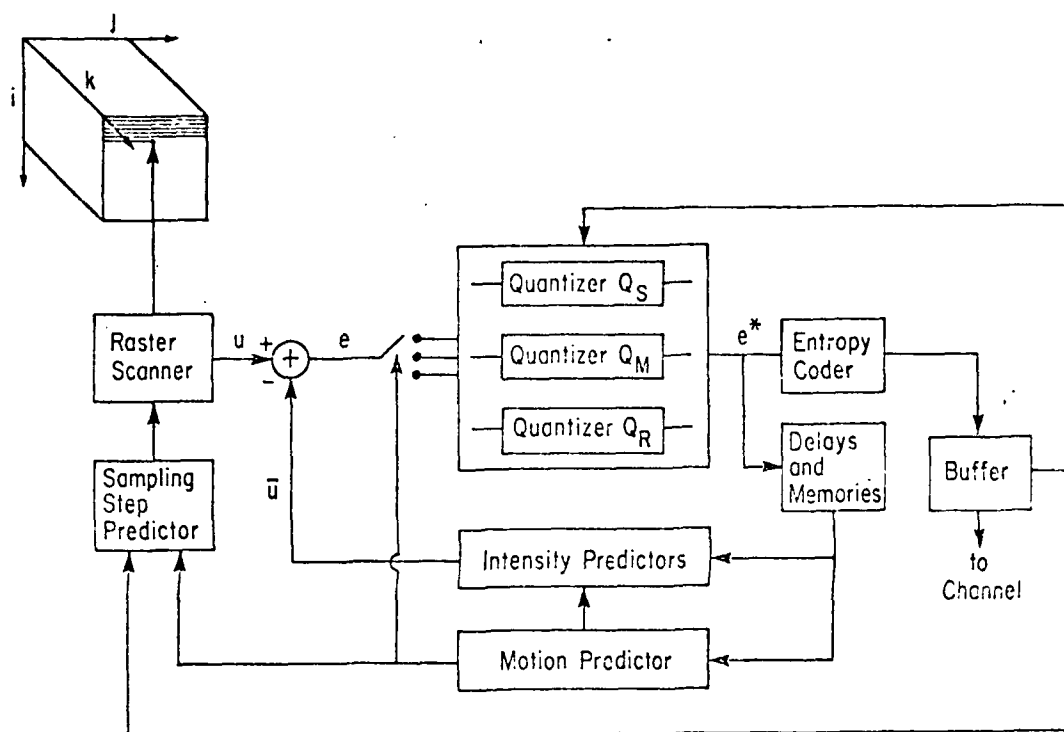


Figure 3-4: An Adaptive Classification Prediction Scheme.

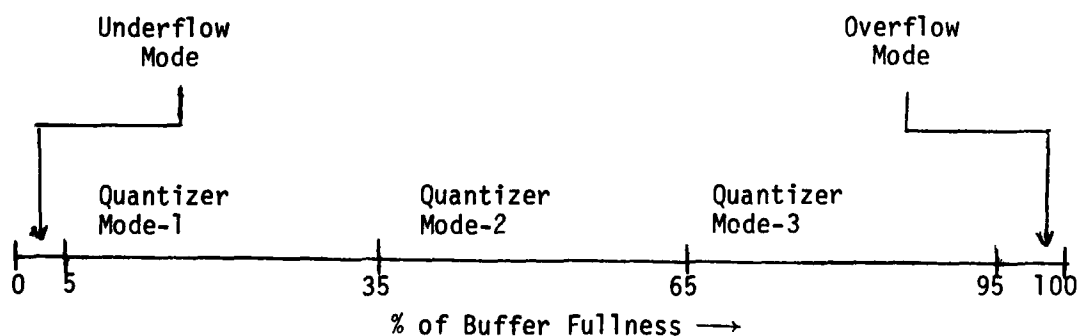


Figure 3-5: Buffer Control Levels for the Adaptive Classification Prediction Scheme.

interframe difference is a good measure of the extent of the motion.

In order to classify motion we measure, over a neighborhood of the pixel, a weighted sum of the absolute interframe difference. A small neighborhood would be sensitive to the quantization error, noise, etc., and a large neighborhood requires more computations and would not respond quickly to the changes in motion. Keeping that in mind, the neighborhood of Figure 3-2 was chosen. Also we have chosen the absolute value of the interframe difference signal as opposed to its square to reduce the sensitivity to large quantization errors. Based on these criteria, we define a neighborhood activity index $A_{k,i,j}$ as

$$A_{k,i,j} = \sum_{(x,y) \in \mathcal{N}} w_{x,y}^a |u_{k,i+x,j+y}^* - u_{k-1,i+x,j+y}^*| \quad (3-1)$$

where $w_{x,y}^a$ are the weights and \mathcal{N} the pixel neighborhood, and

$$w_{x,y}^a \geq 0 \quad (3-2)$$

$$\mathcal{N} = \{(0,-p); (-1,-1); (-1,0); (-1,1)\} \quad (3-3)$$

and

$$p = \begin{cases} 2, & \text{if coder is in 2:1 sub-sampling mode} \\ 1, & \text{otherwise.} \end{cases} \quad (3-4)$$

Then $u_{k,i,j}$ is assigned to one of the three classes-- c_S (stationary), c_M (slow motion) or c_R (rapid motion) as follows.

$$u_{k,i,j} \in \begin{cases} c_S, & \text{if } A_{k,i,j} \leq L_1 \text{ or } A_{k,i,j} \leq L_2 \text{ and } u_{k,i,j-1} \in c_S \\ c_R, & \text{if } A_{k,i,j} \leq L_4 \text{ or } A_{k,i,j} \leq L_3 \text{ and } u_{k,i,j-1} \in c_R \\ c_M, & \text{otherwise} \end{cases}$$

where $L_1 \leq L_2 \leq L_3 \leq L_4$ are predetermined thresholds. Note that we have chosen elastic thresholds between the classes. In order to jump from c_S to c_M or c_R we have a higher threshold (L_2) than to jump from c_M or c_R to c_S , L_1 . The converse is true for transition from and to class c_R . This reduces the sensitivity of classification to quantization and other noise.

3.2.2 Intensity Prediction - Based on the classification of motion, the prediction of $u_{k,i,j}$, denoted by $\bar{u}_{k,i,j}$, is given by

$$\bar{u}_{k,i,j} = \begin{cases} u_{k-1,i,j}^*, & \text{if } u_{k,i,j} \in c_S \\ u_{k,l,i-q,j-r}^*, & \text{if } u_{k,i,j} \in c_M \\ (\rho_j)^p u_{k,i,j-p}^* + \rho_i u_{k,i-1,j}^* - \rho_i (\rho_j)^p u_{k,i-1,j-p}^*, & \text{if } u_{k,i,j} \in c_R \end{cases} \quad (3-5)$$

where ρ_i and ρ_j are one step correlation coefficients along i and j respectively, and the pair $(q,r) \in \mathcal{M}$ is chosen so as to minimize

$$B_{k,i,j}(q,r) = \sum_{(x,y) \in \mathcal{M}} w_{x,y}^b |u_{k,i+x,j+y}^* - u_{k-1,i+x,q,j+y+r}^*|,$$

where

$$w_{x,y}^b \geq 0 \quad (3-6)$$

and

$$\mathcal{M} = \{(s,t); s,t = -1,0,1; (s,t) \neq (0,0)\}$$

and N and p are given by (3-3) and (3-4). The above simply means that for class c_M we search the direction of the nearest neighbors of Figure 3-3 which minimizes the index $B_{k,i,j}$. Note that for a pixel classified as having rapid motion we have used a 2-D causal spatial predictor based on the separable first order Markov covariance model.

3.2.3 Subsampling - The fact that in moving areas spatial resolution can be traded off for temporal resolution [57] could be utilized to achieve more compression by subsampling the images in the moving area in conjunction with the buffer contents. The intensity of the subsampled pixels is obtained by linear interpolation.

3.2.4 Quantization and Coding - The prediction errors for different classes are quantized using different quantizers. In order to achieve a rate very close to the entropy of the quantizer output symbols, a group of quantizer symbols of fixed length is coded at a time and a binary code is generated using the Huffman coding algorithm [1].

3.2.5 Buffer Length Control - To limit the buffer requirement to a reasonable size, the quantizer levels are changed as a function of the numbers of bits residing in the buffer. The levels of the quantizers are changed so as to decrease the entropy of the output symbols as the buffer contents increase and vice versa. On the average the entropy of each of the quantizers is matched to that of the desired transmission rate (or compression ratio). To prevent buffer overflow or underflow, we use the same technique as used in [13] and also described in section 3.1.

3.2.6 Simulation Parameters - We have simulated the above scheme to achieve a compression ratio ($C \cdot R \cdot$) of 16 or a bit-rate of .5 bit/pixel

for the Head and Shoulders data. The buffer length was chosen to store one coded frame at this rate, i.e., $.5 \times 256 \times 256$ bits. The weights of (3-2) and (3-6) were all selected to be unity except that $w_{0,-p}^a$ was chosen to be two. This was done since this is the only pixel corresponding to the current line and is known to be not interpolated. Of the rest three pixels, which belong to the previous line, as many as two could be interpolated pixels, which have higher errors.

The value of the classification thresholds were determined experimentally to minimize the prediction error (for each class) and were found to be $L_1 = 10$, $L_2 = 14$, $L_3 = 50$ and $L_4 = 70$. However, they could also be found by finding the expected value of $A_{k,i,j}$ for transition from one class to another. This would require the knowledge of covariance function of the image and the probability distributions of the interframe difference signal, the interframe motion and the quantizer noise.

To exchange temporal and spatial resolution, a 2:1 subsampling was done for classes c_M and c_R . For simplicity, the values of ρ_i and ρ_j in (3-5) were chosen to be unity. The buffer control levels are shown in Figure 3-5.

The input and output levels of various quantizers are shown in Table 3-1. We have chosen a set of alphabets $\mathcal{C} = \{c_1, c_2, c_3\}$ for the output of the quantizer Q_S . As we are doing a 2:1 subsampling for classes c_M and c_R , we have decided to choose the symbols for Q_M and Q_R from $\mathcal{C} \times \mathcal{C}$ to simplify the design of the binary encoder. Wherever there are more than one symbols available for an output, only one is sent at a time and each takes its turn in a fixed order, e.g., the code c_1c_1/c_1c_3 means that the first time c_1c_1 is transmitted and the next time c_1c_3 is transmitted. Then again the cycle is repeated. At the decoder, both of

TABLE 3-1
QUANTIZATION FOR ADAPTIVE CLASSIFICATION PREDICTION SCHEME

QUANTIZER	DECISION LEVELS			OUTPUT LEVELS			PROBABILITY	SYMBOLS
	MODE-1	MODE-2	MODE-3	MODE-1	MODE-2	MODE-3		
Q_S	$\bar{+2}$	$\bar{+5}$	$\bar{+6}$	-5 0 5	-8 0 8	-9 0 9	.04 .92 .04	c_1 c_2 c_3
	$\bar{+5}$ $\bar{+11}$	$\bar{+7}$ $\bar{+12}$	$\bar{+9}$ $\bar{+15}$	-17 -8 0 8 17	-18 -10 0 10 18	-21 -12 0 12 21	.0032 .074 .846 .074 .0032	$c_1 c_1 / c_1 c_3$ $c_1 c_2 / c_2 c_1$ $c_2 c_2$ $c_3 c_2 / c_2 c_3$ $c_3 c_1 / c_3 c_3$
	$\bar{+8}$	$\bar{+10}$	$\bar{+12}$	-20 0 20	-22 0 22	-24 0 24	.077 .846 .077	$c_1 c_2 / c_2 c_1$ $c_2 c_2$ $c_3 c_2 / c_2 c_3$

these codes are decoded into the same output level of the quantizer. The entropy of the symbol set is .48 bits. We code a group of four alphabets at a time using the Huffman code [1] and achieve an average rate of .52 bit/alphabet, which is very close to the entropy and also our desired bit-rate of .5 bit/pixel, as each pixel generates one alphabet.

3.3 Results and Comparisons:

Table 3-2 shows the performance of the frame replenishment cluster coding scheme for the Head and Shoulders data. It is assumed that the first frame is available without any distortion at the receiver as well as the transmitter. The stationary area corresponds to those pixels of a frame which are classified as insignificant changes from the previous frame and the moving area corresponds to the significant changes. Buffer overflow area corresponds to the area of a frame which is repetition of the previous frame after the buffer contents exceed point F in Fig. 3-1.

Figures 3-6(a) and 3-6(b) show the resulting images at bit-rates of .5 bit/pixel and 1 bit/pixel, respectively, assuming no transmission channel errors. As is evident from Table 3-2, the scheme performs poorly at .5 bit/pixel since about 60% of the time the contents of the previous frame are repeated. Thus, the motion would be reproduced with jerkiness. The temporal lag is evident from Fig. 3-6(a) where the areas of the image correspond to different frames in the original data.

The performance of the adaptive classification prediction scheme is shown in Table 3-3 and Figure 3-6(c). Although the SNR is much better than the cluster coding scheme, there is visibly poor spatial resolution in the moving areas.

TABLE 3-2

FRAME REPLENISHMENT CLUSTER CODING RESULTS FOR HEAD AND SHOULDERS DATA.

IMAGE AREA	.5 Bits/Pixel		1 Bit/Pixel	
	% OF TOTAL AREA	S/N	% OF TOTAL AREA	S/N
Stationary	29.40	38.6 dB	68.66	39.09 dB
Moving	11.38	31.19 dB	23.61	30.14 dB
Buffer-Overflow	58.52	25.90 dB	6.61	25.48 dB
TOTAL	100.00	27.88 dB	100.00	33.0 dB

TABLE 3-3

ADAPTIVE CLASSIFICATION PREDICTION CODING OF HEAD AND SHOULDER DATA,
BIT RATE = 0.5.

IMAGE AREA	% OF TOTAL AREA	S/N
Stationary	55.05	39.09 dB
Slow Motion	37.15	33.97 dB
Rapid Motion	7.76	27.77 dB
TOTAL	100.00	34.75 dB



(a) Frame Replenishment Cluster Coding.
 Bit-rate = .5 bit/pixel, SNR = 28.23 dB.



(b) Frame Replenishment Cluster Coding.
 Bit-rate = 1 bit/pixel, SNR = 34.19 dB.



(c) Adaptive Classification Prediction Coding.
 Bit-rate = .5 bit/pixel, SNR = 34.35 dB.

Results of Interframe Predictive Schemes for Head and Shoulders Frame 8

Figure 3-6

The schemes described above have the advantage that they require a very low storage capacity for the image data, about 1 image line, as well as have low computational complexity.

CHAPTER IV

TRANSFORM CODING TECHNIQUES

The theory of transform coding of images can be found in [28,31,40, 58,77]. For a brief description see section 1.3.

Figure 4-1 shows a block diagram of a simple 3-D transform coding scheme for multiframe images. For practical reasons of data manipulation and management (in hardware or software), the 3-D data array is first divided into smaller arrays called sub-blocks. Let U be one such array of size $L \times M \times N$ and let V denote its transform. Since for a sub-block size of $L \times M \times N$, L image frames need to be stored, the data array containing L frames will be referred to as a block. Hence further division of this data has been referred to here as a sub-block. The three dimensional discrete unitary linear transforms that we consider are separable in the three dimensions, analogous to the three dimensional Fourier transform of a continuous function $f(x,y,z)$, viz.,

$$\begin{aligned} F(\omega_1, \omega_2, \omega_3) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y,z) e^{-j(\omega_1 x + \omega_2 y + \omega_3 z)} dx dy dz \\ &= \int_{-\infty}^{\infty} \left\{ \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(x,y,z) e^{-\omega_1 x} dx \right] e^{-\omega_2 y} dy \right\} e^{-\omega_3 z} dz . \end{aligned}$$

For the discrete array U , $\{u(k,i,j); 1 \leq k \leq L, 1 \leq i \leq M, 1 \leq j \leq N\}$, the analogous 3-D transformation is

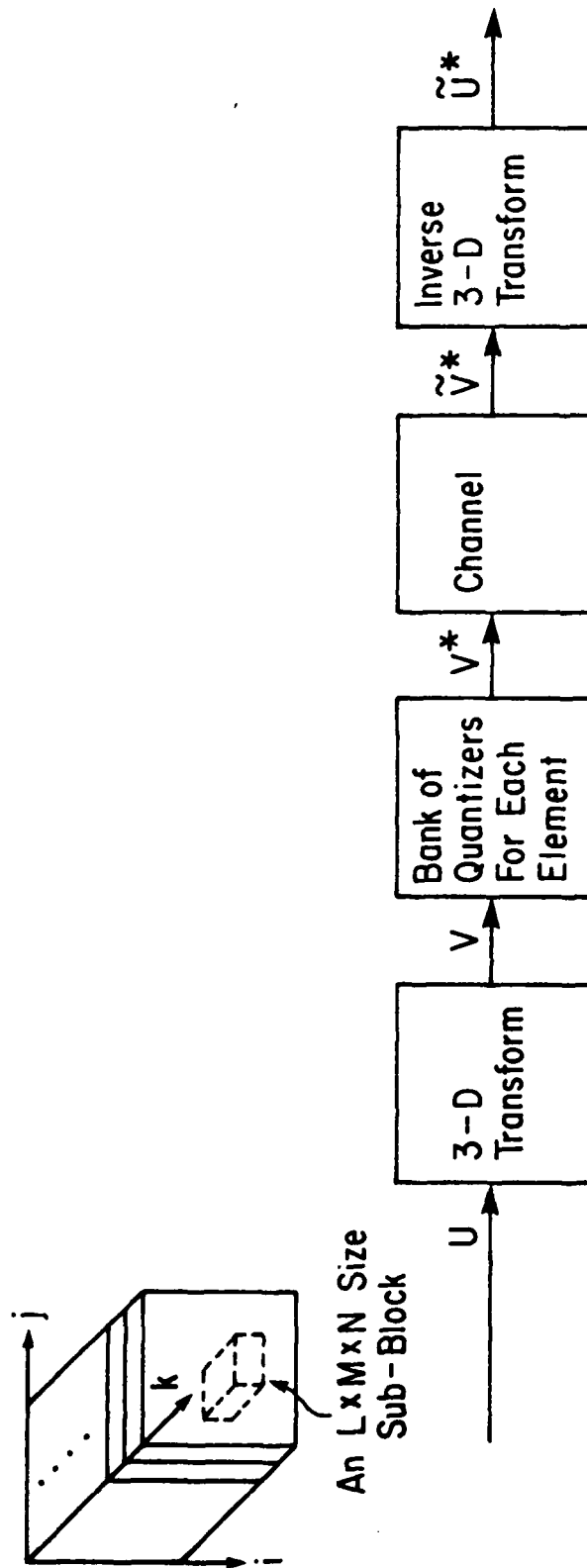


Figure 4-1: An Interframe Transform Coding Scheme.

$$v(\ell, m, n) = \sum_{k=1}^L \sum_{i=1}^M \sum_{j=1}^N u(k, i, j) \psi_L(\ell, k) \psi_M(m, i) \psi_N(n, j)$$

$$1 \leq \ell \leq L, 1 \leq m \leq M, 1 \leq n \leq N \quad (4-1a)$$

where ψ_M is an $M \times M$ transform matrix for an $M \times 1$ vector. Because of the separability of this transformation in each dimension, (4-1a) can be written as a sequence of three one dimensional transformations

$$\begin{aligned} v_1(\ell, i, j) &= \sum_{k=1}^L u(k, i, j) \psi_L(\ell, k); \quad 1 \leq i \leq M, 1 \leq j \leq N, 1 \leq \ell \leq L \\ v_2(\ell, m, j) &= \sum_{i=1}^M v_1(\ell, i, j) \psi_M(m, i); \quad 1 \leq \ell \leq L, 1 \leq j \leq N, 1 \leq m \leq M \\ v_3(\ell, m, n) &= \sum_{j=1}^N v_2(\ell, m, j) \psi_N(n, j); \quad 1 \leq \ell \leq L, 1 \leq m \leq M, 1 \leq n \leq N \end{aligned} \quad (4-1b)$$

$$v(\ell, m, n) \equiv v_3(\ell, m, n) .$$

For an arbitrary ψ , the number of operations would be $LMN(L+M+N)$. If ψ is a fast transform, such as the fast Fourier transform (FFT) [8], then the operation count is reduced to the order of $LMN \cdot \log_2(LMN)$.

Each sample of the transform array, called transform coefficient, is generally quantized independently by a zero memory quantizer. The overall coder efficiency is maximized (with respect to the mean square error criterion) when the transform coefficients are uncorrelated (which is a property of the optimum Karhunen-Loeve transform [40,76]). The quantizer design depends on the probability distribution of the transformed samples. Experimentally, for the Cosine transform, the samples $v(\ell, m, n)$ have been modeled quite well for image data by the Laplacian density model, Roese [67],

$$p(x) = \frac{1}{\sqrt{2}\sigma} \exp\left\{-\frac{\sqrt{2}|x-\mu|}{\sigma}\right\} \quad (4-2)$$

where μ and σ are the mean and the standard deviation of the random variable x . Many times, the first sample $v(1,1,1)$ is modeled by the Rayleigh density [67]. This is because image data is often non-negative and, for many transforms, the first sample is proportional to the average (or the so called d.c.) value of the data and is therefore non-negative. Hence $v(1,1,1)$ would be a non-negative random variable. However, if the data has been modeled by zero mean random process (or has been converted to be such by subtracting the mean from the data) then the Laplacian density model for $v(1,1,1)$ suffices. In the sequel, without loss of generality, we will assume $u(k,i,j)$ and $v(\ell,m,n)$ to be zero mean random variables. Let

$$\sigma_v^2(\ell,m,n) \triangleq E[v^2(\ell,m,n)] \quad (4-3)$$

be the variance of the transform coefficient $v(\ell,m,n)$. In Fig. 4-1, if there are no channel errors in storage or readout of the quantized samples, we will have $\tilde{v}^*(\ell,m,n) = v^*(\ell,m,n)$. The average mean square distortion between the input and output is defined as

$$D = \frac{1}{LMN} \sum_{k=1}^L \sum_{i=1}^M \sum_{j=1}^N E[\{\tilde{u}^*(k,i,j) - u(k,i,j)\}^2] . \quad (4-4)$$

Since the transformation is unitary, Parseval's relation implies

$$D = \frac{1}{LMN} \sum_{\ell} \sum_m \sum_n E[\{\tilde{v}^*(\ell,m,n) - v(\ell,m,n)\}^2] .$$

and in the absence of channel errors

$$D = \frac{1}{LMN} \sum_{\ell} \sum_m \sum_n E[\{v^*(\ell, m, n) - v(\ell, m, n)\}^2] .$$

If we define the quantizer characteristics as

$$q(x) = \begin{array}{l} \text{mean square quantizer error for a unit} \\ \text{variance input random variable quantized} \\ \text{to } x \text{ bits} \end{array} \quad (4-5)$$

we can write

$$D = \frac{1}{LMN} \sum_{\ell} \sum_m \sum_n \sigma_v^2(\ell, m, n) q(b_{\ell, m, n}), \quad (4-6)$$

where $b_{\ell, m, n}$ = number of bits allocated to the coefficient $v(\ell, m, n)$.

Now, if the total number of bits available is fixed, i.e.,

$$\sum_{\ell} \sum_m \sum_n b_{\ell, m, n} = LMN b \quad (4-7)$$

where b = average bit rate in bits per sample, the overall distortion is minimized by finding the optimal bit allocation among the various samples such that the distortion D given by (4-6) is minimized. Since

$$b_{\ell, m, n} \geq 0 \quad (4-8)$$

(4-6) is to be minimized subject to the constraints of (4-7) and (4-8).

Another desirable constraint is to require

$$b_{\ell, m, n} = \text{integer} . \quad (4-9)$$

The above minimization can be performed by a simple integer programming algorithm, originally due to Fox [89]. Jain and Wang [86] have applied

this algorithm for finding integer bit allocation in hybrid coding of intraframe images for several practical quantizers. (For other methods of approximate bit allocation see [28,70]). Once the bit allocations are known, each transform sample $v(\ell, m, n)$ is quantized to $b_{\ell, m, n}$ bits.

As we can see from the above analysis, a transform coder design requires the knowledge of the transform coefficient variances, $\sigma_v^2(\ell, m, n)$. For a simple transform coding scheme with a fixed quantizer for each of the transform coefficients, the multiframe images are assumed to be wide sense stationary (although, as pointed out in chapter II, this is a very poor approximation for the motion images). With this assumption, it can be easily seen that $\sigma_v^2(\ell, m, n)$ can be obtained by the knowledge of the covariance function of the array U (see [49,67]). One approach has been to model the covariance by some simple function, e.g., as a product of first order stationary Markov process covariance models [67] defined by

$$c_{\ell, m, n} \triangleq E[u(k', i', j') \cdot u(k' + \ell, i' + m, j' + n)] \quad (4-10a)$$

$$= \sigma^2 \cdot \rho_k^{\ell} \cdot \rho_i^m \cdot \rho_j^n \quad (4-10b)$$

where σ^2 is the variance of the data sample $u(k, i, j)$ and the ρ_k , ρ_i , and ρ_j are the one step correlation parameters along the indices k , i , and j , respectively. An alternative approach is to measure the covariance function on a portion of the data [49] similar to the intraframe case as in (4-5) and use these for the rest of the data. However, the transform domain statistics, i.e., $\sigma_v^2(\ell, m, n)$, can be directly estimated from the transform coefficients. In this case the estimate is given by

$$\sigma_v^2(\ell, m, n) = \frac{1}{B_0} \sum_v v^2(\ell, m, n) \quad 1 \leq \ell \leq L, \quad 1 \leq m \leq M, \quad 1 \leq n \leq N, \quad (4-11)$$

where B_0 is the number of data sub-blocks used in the calculation of (4-11) and the summation is taken over all these sub-blocks.

The array of quantized transform coefficients, V^* , is encoded and transmitted or stored. Its received value, \tilde{V}^* , is inverse transformed (by interchanging U and V and replacing Ψ with Ψ^{*T} in (4-1)) to obtain the reproduced value of U as \tilde{U}^* . In the coding experiments of this chapter and the next chapter we assume the channel to be noise free, i.e.,

$$V^* = \tilde{V}^* \Rightarrow U^* = \tilde{U}^* .$$

For the discussion on coding for noisy channels, refer to chapter VI.

4.1 Adaptive Interframe Transform Coding Schemes

In the transform coding scheme discussed above, the multiframe images were modeled as a 3-D stationary random process (in the wide sense). In reality, the multiframe motion images are nonstationary, in general. The nonstationarity exists in the spatial as well as the temporal dimensions, and the latter appears to be more severe between the two. This is because the temporal direction is deterministically related (except for the noise due to camera & digitization) to the spatial coordinates as discussed in chapter II. Hence, a stationary random process characterization is not a realistic assumption. The nonstationarity in the spatial domain is mainly due to the presence of sharp object edges (or features) within an image frame.

The encoding of multiframe motion images with the assumption of stationarity, therefore, results in large degradations in the sharp features within a frame, and in the reproduction of motion features. Usually, these are the more desirable features.

Some researchers have developed methods of coding single frame images by separating the nonstationarities (such as edges) and coding them separately [75,83]. The residual image (after separating or subtracting edges) is then modeled accurately by a stationary process and can be coded using a simple transform coding scheme such as described above (the intraframe or 2-D transform coding is a special case of the interframe scheme with $L = 1$). However, these methods result in increased complexity and their extension to the interframe (or 3-D) transform coding seems difficult.

We have investigated the possibility of some simple extensions of transform coding which would improve its performance by accounting, in some way, for the nonstationarity. We have found that the concept of "activity index" proposed by Gimlet [21] for intraframe transform coding can be extended to the interframe coding by finding a modified activity index. As we have seen in chapter II, the interframe variance (or IFV) is a good measure of the combined spatial and temporal activity between two successive frames of multiframe motion images. Thus, an average of the IFV measured over a sub-block between each pair of successive frames, given by

$$a = \frac{1}{(L-1)MN} \sum_{k=2}^L \sum_{i=1}^M \sum_{j=1}^N \{u(k,i,j) - u(k-1,i,j)\}^2 \quad (4-12)$$

could be used as a good measure for the activity index.

In [21] the adaptation is achieved by classifying a two dimensional sub-block into one of the 4 classes, based on the value of the activity index (which is nothing but the variance of the sub-block), by a threshold classifier. The thresholds are chosen such that each class has equal

occurrences in an image frame. Each class is assigned different but fixed number of total bits (obviously a class having higher activity index is assigned more bits). Thus this scheme operates at fixed bit-rate per frame. We would like to point out that what is essentially being achieved by this adaptation is to approximate a nonstationary process by 4 piecewise stationary processes. Also note that for each class the thresholds would vary from one image frame to another with the distribution of motion. Thus, making the piecewise stationary approximation poorer because, for the same class, the range of activity index is no longer fixed. Therefore, we have chosen fixed thresholds for classification.

We also choose 4 classes. The selection of number of classes is a trade-off between performance and complexity. The classification for each sub-block is coded using 2 bits. For each class separate bit-rates and statistics are used. Once the statistics for each class are known (or measured) the bit-rates could be determined from the distortion-rate curves by fixing the distortion level for each class. The activity index (or IFV) thresholds for classification depend on the nature of the data and the sub-block size. Their suitable values can be found from the histogram of the activity index.

In the adaptive scheme described above, both the bit-rate and statistics were adapted for each class. However, if desired, one of them could be kept constant at the cost of only a partial improvement over the usual (or non-adaptive) scheme.

4.2 Experimental Results

The adaptive and non-adaptive interframe transform coding methods of the previous sections as well as the usual (or non-adaptive) intraframe

transform coding were applied to some of the data sets described in chapter I.

Due to the superior performance of the Cosine transform for data compression of highly correlated data (see appendix B and [33,49,67]) we have chosen the transform matrix Ψ throughout this work as the discrete Cosine transform (or DCT) matrix (see Ahmed et al. [2]) defined as

$$\Psi_M(i,j) = \begin{cases} \sqrt{1/M} & , i = 1, 1 \leq j \leq M \\ \sqrt{2/M} \cos \frac{(2j-1)(i-1)}{2M} & , 2 \leq i \leq M, 1 \leq j \leq M \end{cases} \quad (4-13)$$

4.2.1 Head and Shoulders Images - All the three methods are compared for this data set. Since it contains motion images with motion being localized in certain areas of an image frame, it is a good candidate for comparing the effect of adaptations. For interframe (or 3-D transform) coding a sub-block size of $16 \times 16 \times 16$ was chosen as in [67]. To compare the performance of interframe and intraframe schemes, two sub-block sizes for intraframe transform coding were chosen. The first, 16×16 , is used to compare the contribution of the temporal redundancy exploited by the interframe scheme. The second size, 64×64 , is used to have the same number of samples in the intraframe and interframe sub-blocks. Since the performance of a scheme also depends on the knowledge of the statistics, some of the statistical models are also compared.

For intraframe coding we compare three statistical models--(i) Separable covariance model of (A-1) with $\rho_1 = \rho_j = .95$; (ii) Measured statistical model, which is obtained by suppressing index l in (4-11), given in Table A-1; and (iii) Isotropic covariance model with correction given in Table A-4 (see appendix A for discussion on modeling intraframe statistics).

Table 4-1 gives the performance of these models for a sub-block size of 16×16 . As expected, the performance of the separable model is the worst and that of the measured statistical model is the best. The isotropic model with correction is closer to the measured statistical model at low bit-rates and in between at higher bit-rates. Also, the superior performance of the measured statistical model increases with the bit-rate. This is also expected, since at lower bit-rates only the low order (or low spatial frequency) transform coefficients are transmitted (from the distortion-rate considerations) and usually the simple parametric models such as the separable and the isotropic (without correction) do well in predicting their statistics (as can be seen by comparing Tables A-1 and A-3).

Table 4-2 shows the performance of the intraframe scheme for the sub-block size of 64×64 for two of the models (except the separable, which is expected to do relatively worse for higher sub-block size). We notice an improvement between 1-2 dB for the isotropic model and about 3-4 dB for the measured statistical model over the 16×16 case. Thus, the relative superior performance of measured statistical model increases with the increase in the sub-block size. This again is expected.

Figures 4-2 and 4-3 show some of the images corresponding to frame #8 resulting from the intraframe transform coding. In general, the relative visual quality for various models and array sizes is in agreement with the mean square performance. At low SNR (below 35 dB), the noise in the background areas is quite visible in addition to the blurring of sharp features (or edges). For high SNR (above 37 dB) the visual quality is good.

TABLE 4-1

SNR FOR NON-ADAPTIVE INTRAFRAME COSINE TRANSFORM CODING OF THE HEAD AND SHOULDERS IMAGES FOR THREE STATISTICAL MODELS. SUB-BLOCK SIZE = 16×16 .

S. N.	BIT-RATE PER PIXEL	STATISTICAL MODEL		
		SEPARABLE	ISOTROPIC WITH CORRECTION	MEASURED STATISTICS
1	.25	28.02 dB	29.76 dB	30.29 dB
2	.50	30.03 dB	33.41 dB	34.16 dB
3	1.00	35.47 dB	37.41 dB	39.63 dB
4	2.00	40.92 dB	42.38 dB	45.52 dB

TABLE 4-2

SNR FOR NON-ADAPTIVE INTRAFRAME COSINE TRANSFORM CODING OF THE HEAD AND SHOULDERS IMAGES FOR TWO STATISTICAL MODELS. SUB-BLOCK SIZE = 64×64 .

S. N.	BIT-RATE PER PIXEL	STATISTICAL MODEL	
		ISOTROPIC WITH CORRECTION	MEASURED STATISTICS
1	.25	31.49 dB	33.29 dB
2	.50	35.31 dB	38.06 dB
3	1.00	39.27 dB	43.71 dB
4	2.00	43.44 dB	48.49 dB

TABLE 4-3

SNR FOR NON-ADAPTIVE INTERFRAME COSINE TRANSFORM CODING OF THE HEAD AND SHOULDERS IMAGES FOR TWO STATISTICAL MODELS. SUB-BLOCK SIZE = $16 \times 16 \times 16$.

S. N.	BIT-RATE PER PIXEL	STATISTICAL MODEL	
		SEPARABLE	MEASURED STATISTICS
1	.10	28.53 dB	30.72 dB
2	.25	31.31 dB	34.35 dB
3	.50	33.34 dB	37.60 dB
4	1.00	35.49 dB	41.89 dB



(i) Bit-rate = .5 bit/pixel, SNR = 31.11 dB.



(ii) Bit-rate = 1 bit/pixel, SNR = 35.68 dB.

(a) Separable Covariance Model



(i) Bit-rate = .5 bit/pixel, SNR = 33.65 dB.

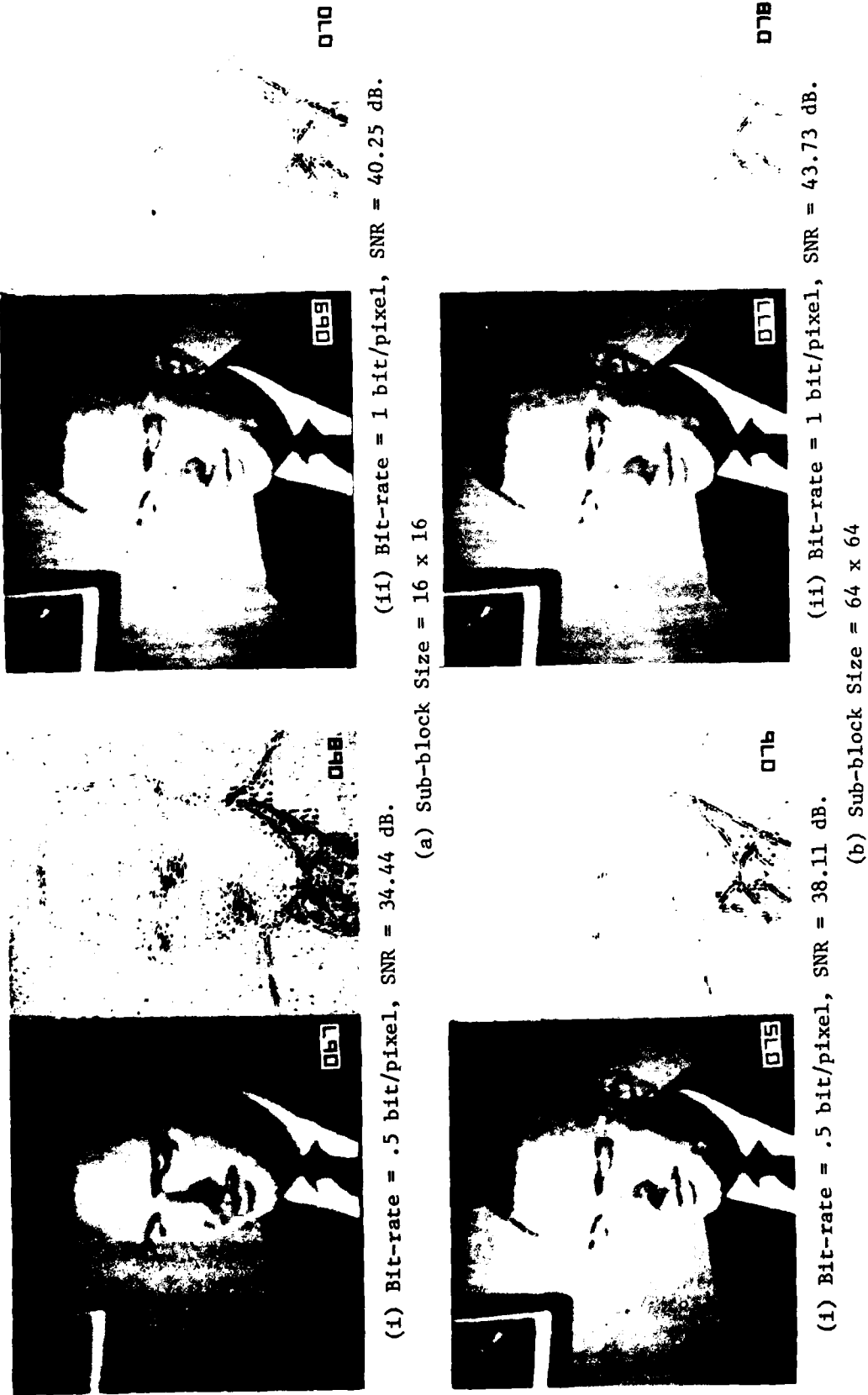


(ii) Bit-rate = 1 bit/pixel, SNR = 37.76 dB.

(b) Isotropic Covariance Model with Correction

Intraframe Transform Coding using Separable and Isotropic Models, Sub-block Size = 16 x 16

Figure 4-2



Intraframe Transform Coding using Measured Statistics in Transform Domain with Various Sub-block Sizes.

Figure 4-3

Table 4-3 shows the performance of the non-adaptive interframe transform coding method for the separable model of (4-10b) with $\rho_i = \rho_j = \rho_k = .95$, and the measured statistical model of (4-11). Tables 4-4 and 4-5 show the bit allocation for these two models. On comparing these two tables we notice that the separable model wastes a large number of bits on high spatial frequencies which contain negligible mean square energy. Thus, it is a poor model (as pointed out earlier) in predicting the variance of high spatial frequencies.

On comparing Tables 4-1 and 4-3 we note that for the separable models, the gains due to temporal redundancy are only realized at low bit-rates (once again, for the same reason as in the intraframe), and at 1 bit/pixel there are no practical gains. While for the measured statistical models, there are gains of 2-4 dB arising from temporal redundancy, the gains decreasing with increasing bit-rates.

The comparison of Tables 4-2 and 4-3 show that, for the measured statistical models, the gains achieved by the exploitation of the temporal redundancy can be surpassed by an intraframe scheme by simply increasing its sub-block size so that the total sub-block sizes of the interframe and the intraframe schemes are the same. This result, which appears unexpected at first, is because we have modeled the temporal statistics by stationary processes--which is a poor representation in areas of moderate and large motion.

One quantity, to which the relative performance of the intraframe and the interframe schemes for motion images is definitely related, is the amount of motion between successive frames. A lower value of this quantity (resulting in high temporal correlation) will favor the interframe

scheme if all the frames are required to be transmitted. However, adaptations due to motion must be made in areas of significant motion.

Since the intraframe scheme requires much less memory (M image rows) than the interframe scheme (L frames), the above result shows that non-adaptive interframe transform scheme is unattractive from the MSE point of view.

Parts (a) and (b) of Fig. 4-4 show the images corresponding to frame #8 for the non-adaptive interframe transform coding. Comparing Fig. 4-4(a) with Figs. 4-2(a)-(i) and 4-2(b)-(i), we note that for comparable levels of distortions, at a low SNR, the distortion due to the intraframe and the interframe transform coding is differently distributed. The stationary areas are much less noisy (or better reproduced) in the intraframe coding, while the moving area edges are more blurred. This result is expected. Thus, from the point of view of the exchange of spatial and temporal resolution for the motion images, the interframe transform coding method might be more desirable for the same mean square error. This supports the belief that the MSE alone is not a sufficient criterion in comparing various schemes. However, at high SNR values the MSE criterion seems reasonable for comparisons. Figure 4-4(b) shows a significant improvement due to measured statistics over the separable model.

Table 4-6 gives the parameters of the adaptive interframe transform coding scheme. Table 4-7 gives the performance of an adaptive interframe transform coding scheme without adapting the statistics to each class. This was done to separately study the effects of the adaptations of bit-rates and the statistics to the classification. For the first entry of this table even the bit-rates were forced to be the same. Thus, it

AD-A076 841

CALIFORNIA UNIV DAVIS SIGNAL AND IMAGE PROCESSING LAB F/G 9/a
 INTERFRAME ADAPTIVE DATA COMPRESSION TECHNIQUES FOR IMAGES.(U)
 AUG 79 J R JAIN : A K JAIN DAA629-78-S-0206

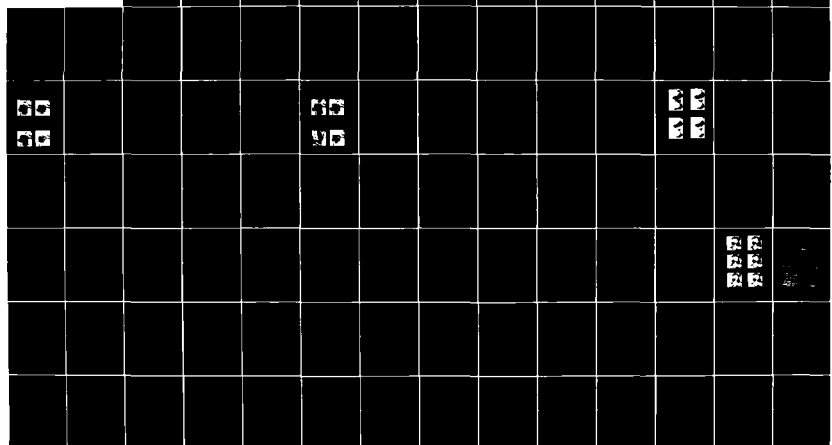
UNCLASSIFIED

ARO-16822.8-2L

NL

2 of 3

2000000





(a) Non-adaptive, Separable Covariance Model,
Bit-rate = .5 bit/pixel, SNR = 32.09 dB.

(b) Non-adaptive, Measured Statistics,
Bit-rate = .5 bit/pixel, SNR = 36.80 dB.



(c) Adaptive, Measured Statistics,
Bit-rate = .1 bit/pixel, SNR = 33.50 dB.

(d) Adaptive, Measured Statistics,
Bit-rate = .5 bit/pixel, SNR = 41.18 dB.

Interframe Transform Coding, Sub-block Size = $16 \times 16 \times 16$

Figure 4-4

TABLE 4-6

PARAMETERS OF 4-CLASS ADAPTIVE TRANSFORM CODING SCHEME FOR HEAD AND SHOULDERS DATA

CLASS NO.	ACTIVITY INDEX	PROBABILITY OF OCCURRENCE
1	0 - 40	.590
2	40 - 100	.172
3	100 - 400	.176
4	400 -	.062

TABLE 4-7

PERFORMANCE OF ADAPTIVE TRANSFORM CODER FOR HEAD AND SHOULDERS DATA. DIFFERENT STATISTICS WERE USED FOR ALL THE CLASSES AND WERE MEASURED OVER ALL THE DATA. BLOCK SIZE IS $16 \times 16 \times 16$.

S. N.	BIT RATE					SIGNAL TO NOISE RATIO IN DECIBELS			
	CLASS #				OVER-ALL	CLASS #			
	1	2	3	4		1	2	3	4
1*	.5	.5	.5	.5	.50	42.56	39.96	35.11	29.17
2	.125	.25	.5	.75	.25	36.15	36.89	35.11	31.85
3	.25	.5	1.0	1.5	.50	39.19	39.96	39.87	36.05
4	.75	1.0	1.5	2.0	1.00	44.84	44.32	43.33	39.17

*This entry corresponds to non-adaptive case and has been given here for comparison.

TABLE 4-8

PERFORMANCE OF ADAPTIVE TRANSFORM CODER FOR HEAD AND SHOULDERS DATA. DIFFERENT STATISTICS WERE USED FOR DIFFERENT CLASSES AND WERE MEASURED OVER EACH CLASS. BLOCK SIZE IS $16 \times 16 \times 16$.

S. N.	BIT RATE					SIGNAL TO NOISE RATIO IN DECIBELS			
	CLASS #				OVER-ALL	CLASS #			
	1	2	3	4		1	2	3	4
1	.031	.094	.25	.375	.10	34.06	33.94	32.60	31.36
2	.125	.25	.50	.75	.25	39.18	37.61	36.40	36.27
3	.25	.50	1.0	1.5	.50	41.34	40.86	41.70	42.88
4	.75	1.0	1.5	2.0	1.00	46.17	45.23	45.48	46.12

corresponds to the non-adaptive case. This was done to see the distribution of the distortion among the classes. Table 4-8 gives the performance of the adaptive scheme for adaptive bit-rates as well as statistics.

From the first entry of Table 4-7 we see that for a non-adaptive case, the average distortion increases with the class number (as expected) and there is about 13 dB difference between class 1 (containing areas of low spatial and temporal activity) and class 4 (containing areas of high spatial and temporal activity). Comparing this with entry 3 of Table 4-7 we see that, for the same average rate, the adaptation of bit-rates alone results in great improvement in the distortion for classes 3 and 4 and in an overall increase of 1.6 dB. From Table 4-8 we see that an additional gain of 2-2.5 dB is achieved by adapting the statistics. Thus the overall improvement for the adaptive scheme over the non-adaptive (interframe) scheme is about 4 dB or a compression gain by a factor of about 2 in addition to the better reproduction of the high spatial activity areas and the motion.

Figure 4-4(c) and 4-4(d) show some images for the adaptive inter-frame transform coding. We can see that the adaptive scheme does far better than the non-adaptive scheme. The performance of the adaptive scheme at .1 bit/pixel is superior to the non-adaptive scheme with the separable model at .5 bit/pixel. This is evident by comparing images (a) and (c) of Figure 4-4, where the former reproduces motion much better (see the lips, the eyes, and the tie). Thus, at low SNR we obtain a compression gain of 5 by the adaptive scheme over the non-adaptive scheme with the separable model and still get better results.

4.2.2 Chemical Plant Images - Since the Chemical Plant images were generated by an airborne camera, the motion is more evenly distributed.

Since any adaptation without motion compensation is not expected to result in significant improvement, only non-adaptive schemes were compared.

Table 4-9 and Figure 4-5 show the results for the measured statistical model. We notice that for these images, the signal-to-noise ratios obtained are much lower than those for the Head and Shoulder images. This is because these images have much lower correlation and the data is more noisy. Comparing the relative performances (Table 4-9), once again, we note that the intraframe scheme at sub-block size of 64×64 is almost as good as the interframe scheme for the sub-block size of $16 \times 16 \times 16$.

4.2.3 X-Ray Projection Images - The projection images in this experiment are the 2-D x-ray projections of a 3-D object at various angles around a fixed axis and do not contain motion. The stationarity assumption is a more valid assumption for these images and thus they are better candidates for interframe transform coding. The given images have a very high correlation between the rows (along i-axis). So a sub-block size of $8 \times 32 \times 16$ was selected.

Table 4-10 shows the performance of the non-adaptive transform coding method with measured statistics. Figure 4-6 shows an original image and some of the coded images at various compression ratios. In Figure 4-7, the mean square error is plotted as a function of frame (or image) number. The periodic occurrence of the error peaks after every eighth frame is due to the fact that these frames lie on the boundaries of our $8 \times 32 \times 16$ sub-blocks. However, this effect diminishes for lower compression ratios. For example, an almost constant mean square error at the compression ratio of 10 is achieved.

TABLE 4-9

SNR FOR NON-ADAPTIVE INTRAFRAME AND INTERFRAME COSINE TRANSFORM CODING
OF THE CHEMICAL PLANT IMAGES WITH MEASURED STATISTICS.

S. N.	BIT-RATE PER PIXEL	INTRAFRAME		INTERFRAME
		SUB-BLOCK SIZE = 16×16	SUB-BLOCK SIZE = 64×64	SUB-BLOCK SIZE = $16 \times 16 \times 16$
1	.5	27.26 dB	28.51 dB	28.65 dB
2	1.0	30.73 dB	32.10 dB	32.16 dB
3	2.0	36.37 dB	38.14 dB	37.97 dB

TABLE 4-10

PERFORMANCE OF THE NON-ADAPTIVE INTERFRAME TRANSFORM CODER FOR THE X-RAY
PROJECTION IMAGES.

S.N.	BIT-RATE PER PIXEL	COMPRESSION RATIO	MEAN SQUARE ERROR	SIGNAL TO NOISE RATIO
1	.04	200	5.522	40.71 dB
2	.125	64	2.153	44.80 dB
3	.25	32	1.277	47.07 dB
4	.50	16	0.757	49.34 dB
5	1.00	8	0.373	52.41 dB
6	2.00	4	0.121	57.31 dB

TABLE 4-9

SNR FOR NON-ADAPTIVE INTRAFRAME AND INTERFRAME COSINE TRANSFORM CODING
OF THE CHEMICAL PLANT IMAGES WITH MEASURED STATISTICS.

S. N.	BIT-RATE PER PIXEL	INTRAFRAME		INTERFRAME
		SUB-BLOCK SIZE = 16×16	SUB-BLOCK SIZE = 64×64	SUB-BLOCK SIZE = $16 \times 16 \times 16$
1	.5	27.26 dB	28.51 dB	28.65 dB
2	1.0	30.73 dB	32.10 dB	32.16 dB
3	2.0	36.37 dB	38.14 dB	37.97 dB

TABLE 4-10

PERFORMANCE OF THE NON-ADAPTIVE INTERFRAME TRANSFORM CODER FOR THE X-RAY
PROJECTION IMAGES.

S.N.	BIT-RATE PER PIXEL	COMPRESSION RATIO	MEAN SQUARE ERROR	SIGNAL TO NOISE RATIO
1	.04	200	5.522	40.71 dB
2	.125	64	2.153	44.80 dB
3	.25	32	1.277	47.07 dB
4	.50	16	0.757	49.34 dB
5	1.00	8	0.373	52.41 dB
6	2.00	4	0.121	57.31 dB



(a) Intraframe, Sub-block Size = 64×64 ,
Bit-rate = .5 bit/pixel, SNR = 28.31 dB.



(c) Intraframe, Sub-block Size = 16×16 ,
Bit-rate = 1 bit/pixel, SNR = 30.60 dB.



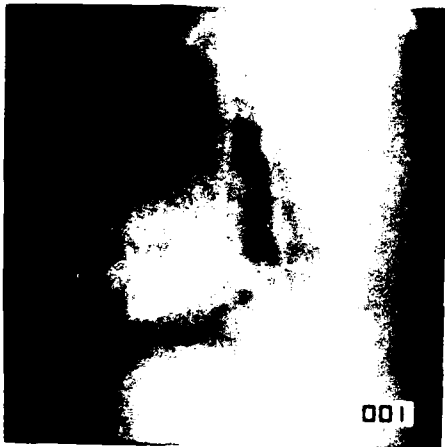
(b) Intraframe, Sub-block Size = 64×64 ,
Bit-rate = 1 bit/pixel, SNR = 31.99 dB.



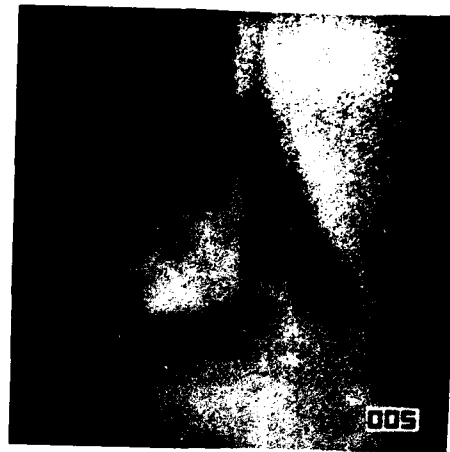
(d) Interframe, Sub-block Size = $16 \times 16 \times 16$,
Bit-rate = 1 bit/pixel, SNR = 31.85 dB.

Intraframe and Interframe Transform Coding using Measured Statistics in Transform Domain.

Figure 4-5



(a) original



(b) C.R. = 200



(c) C.R. = 64



(d) C.R. = 16

Images resulting from data compression of a projection image at angle of view = 0° .

Figure 4-6

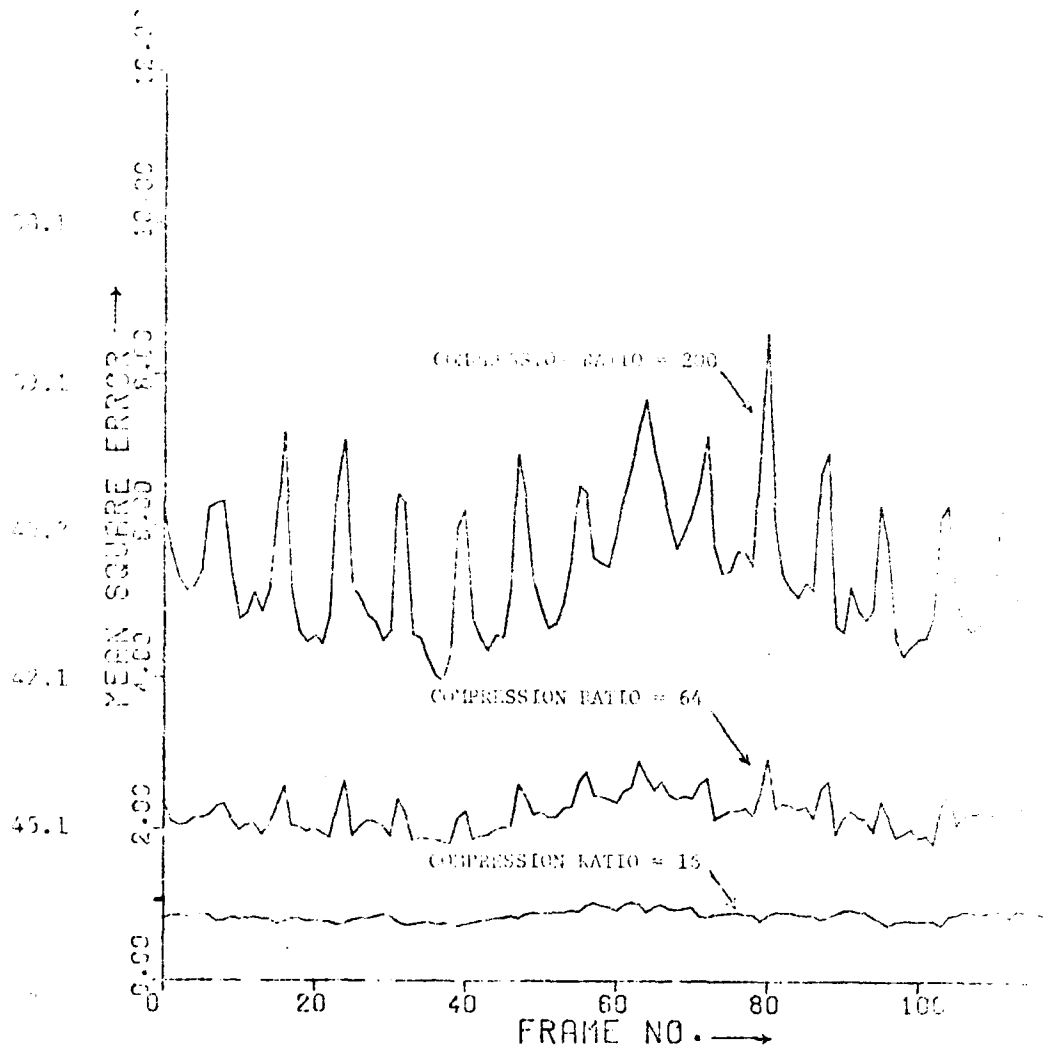
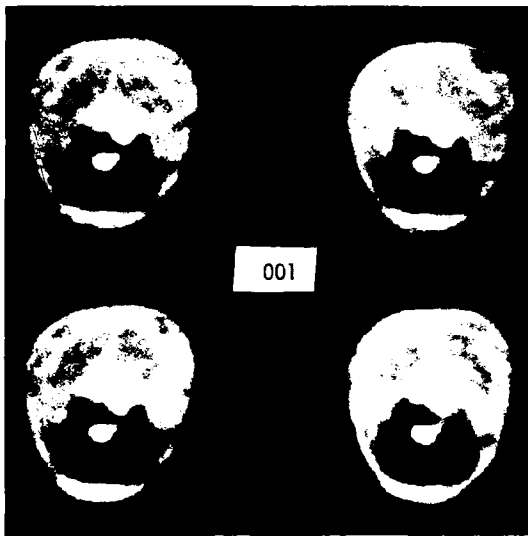


Figure 4-7: Variation of Mean Square Error as a Function of Frame Number for Transform Coding of the X-ray Projection Images.

In the foregoing experiments we have encoded the projection data and shown the coder performance on this data. However, in practice, the medically useful information lies in 3-D view or equivalently, in the multiple adjacent transaxial cross-sections (also called levels) of the object, which are reconstructed from the projection data. The various levels were separately reconstructed by approximating the x-ray cone beam by a sequence of parallel divergent fan beams. This is a reasonable approximation for x-ray cone beam sources far from the object, and permits use of a two dimensional reconstruction algorithm for each level. For our data a divergent beam two-dimensional reconstruction algorithm [27] was used. Final reconstructed images for various levels are of size 64×64 and were displayed after a sample averaging of three adjacent video lines.

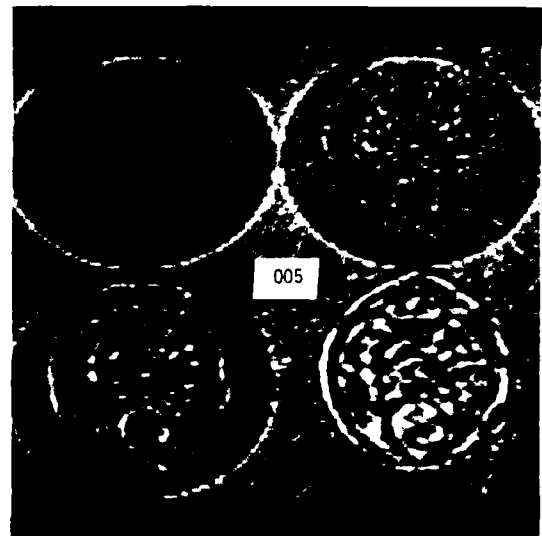
Figures 4-8(a)-(c) show the reconstructed images at levels 34 and 94 (of the total 128 possible) reconstructed from the original as well as from the compressed projection data. Figure 4-8(d) shows the images of the error signal between the original and the compressed reconstructions at various compression ratios for level 94.

The effect of the data compression on resolution is readily observed by viewing the reconstructed images at level 94. Notice the blurring of two dark small circular areas at about the 4 o'clock position (small air passages in the lung called bronchi) as the compression ratio increases. The smaller of the two areas (upper right) starts disappearing at compression ratio of 16, while the other one starts disappearing at 32. Generally the larger features are retained at even higher compression ratios. The error images show that at lower compression ratios the sample-to-sample errors are more or less uncorrelated, while at higher compression ratios the object structure is more visible in the errors.



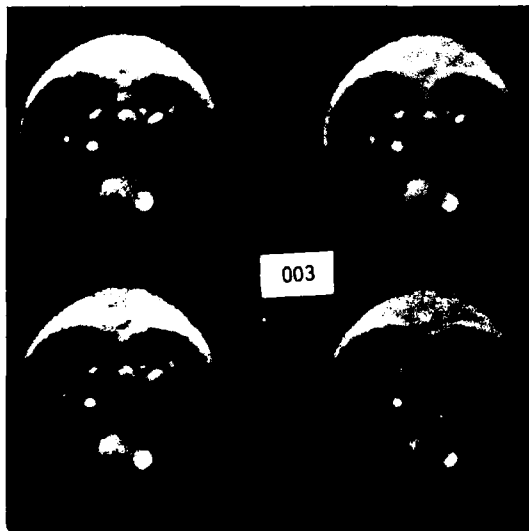
(a) Reconstruction of Level #34

(i) Original	(ii) C.R. = 8
(iii) C.R. = 32	(iv) C.R. = 200



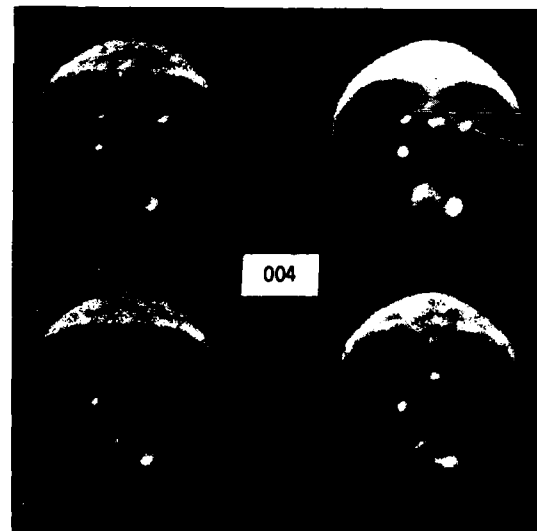
(d) Normalized error in reconstruction of Level #94 due to Data Compression

(i) C.R. = 4	(ii) C.R. = 16
(iii) C.R. = 64	(iv) C.R. = 200



(b) Reconstruction of Level #94

(i) Original	(ii) C.R. = 4
(iii) C.R. = 8	(iv) C.R. = 16



(c) Reconstruction of Level #94

(i) Original	(ii) C.R. = 32
(iii) C.R. = 64	(iv) C.R. = 200

Figure 4-8 : Reconstruction Images

The high values of signal to noise ratio achieved at various compression levels demonstrate that large redundancy is present in the projection images. Preliminary indications are that compression ratios of 8 to 16 are realizable for those applications where very high quality (pixel by pixel) reproduction of the reconstructed images is desired. This would include applications where the reconstructed images are to be used for detection and quantification of objects of small size (e.g., holes in the septum of the heart or distortion of vessels). In other applications, where the medically useful information lies in the size, location and the boundaries of larger objects (e.g., motion of heart wall muscle mass, etc.), larger compression ratios, 64 to 200, may be acceptable.

CHAPTER V

INTERFRAME HYBRID CODING SCHEMES

Because of the rapid changes in the temporal characteristics of motion images, it is desirable to have a predictive coding scheme along the temporal axis. On the other hand, for spatial information transform coding is more efficient. Hybrid coding utilizes the superior performance of transform coding in the spatial domain and the simplicity of DPCM to exploit the temporal correlation with tremendous savings in the memory (requires only a single frame storage for a first order DPCM). Motion compensation methods of chapter II can be successfully employed in this design.

Figure 5-1 shows a simple (or non-adaptive) interframe hybrid coding scheme. First we assume the data to be wide sense stationary. U_k is an $M \times N$ sub-block of the k th frame. V_k is obtained by a 2-D transformation of U_k , and is defined similar to the 3-D transform in (4-1) with $L = 1$. It can also be expressed as

$$V_k = \Psi_M U_k \Psi_N^T.$$

Each transform coefficient is independently coded by DPCM along the temporal axis (or index k) via a suitable autoregressive model representing the statistical characteristics of the data in the temporal direction. In order to limit the storage to one frame, we only consider first order models. For images having piecewise uniform motion from one frame to the next, a first order model would be reasonable. Thus, we have[†]

[†] For any matrix A we denote $A(i,j) \equiv a(i,j)$ to be its (i,j) th element.

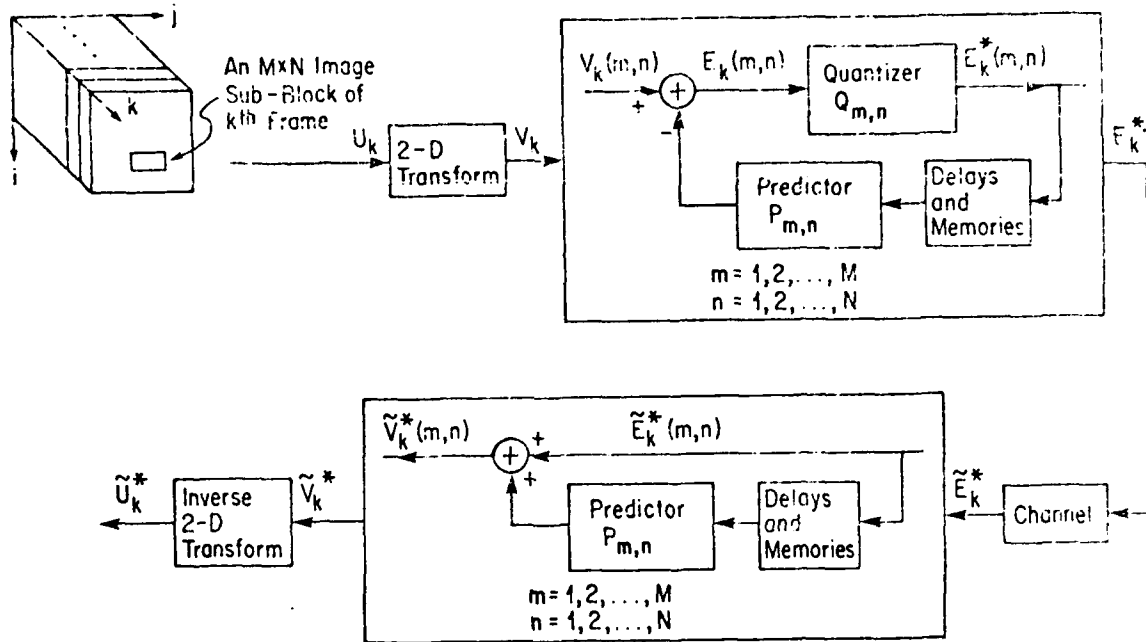


Figure 5-1: A Non-Adaptive Interframe Hybrid Coding Scheme.

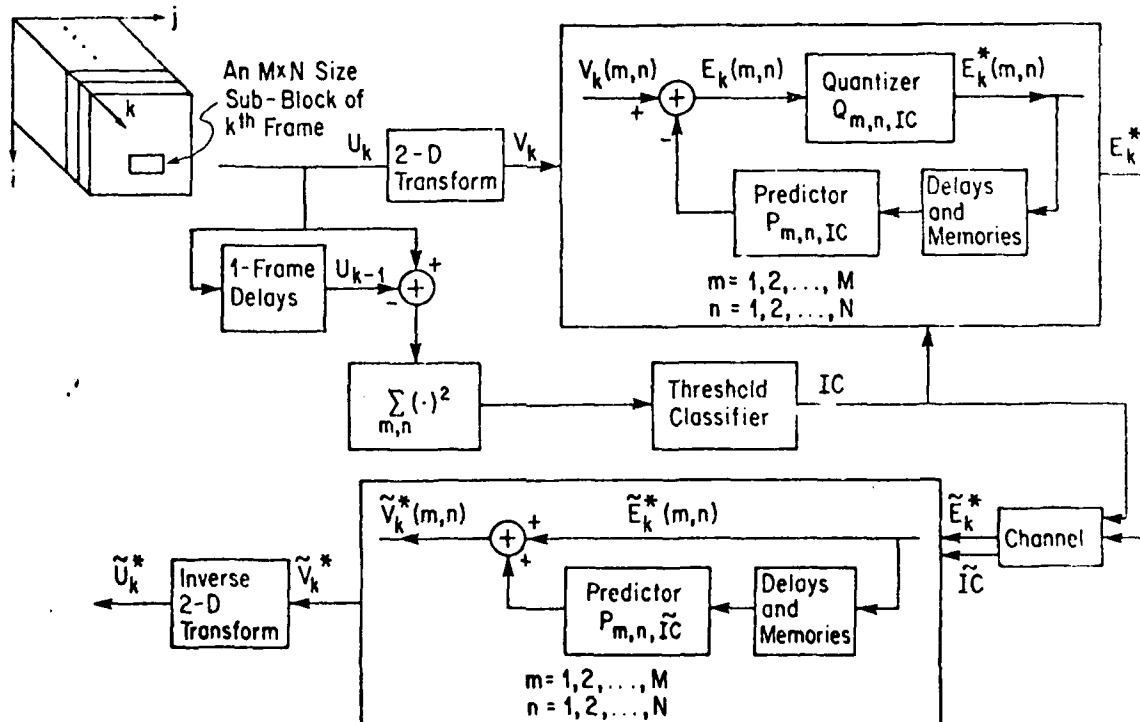


Figure 5-2: An Adaptive Interframe Hybrid Coding Scheme.

$$v_k(m,n) = \alpha_{m,n} v_{k-1}(m,n) + e'_k(m,n), \quad |\alpha_{m,n}| < 1, \quad 1 \leq m \leq M, \quad 1 \leq n \leq N. \quad (5-1)$$

For simplification we assume

$$\alpha_{m,n} = \alpha = \text{Constant}. \quad (5-2)$$

Although the above simplified assumption is not very realistic for motion images, it does not affect the coder performance too adversely (once the assumption of stationarity has been made). This is because at low bit-rates only the low order (or high mean square energy) transform coefficients are transmitted. For these a constant value for the predictor coefficients $\alpha_{m,n}$ given by (5-2) has been found to be adequate. With references to Figure 5-1, the various predictive coding equations, for each $1 \leq m \leq N, 1 \leq n \leq N$, are given as follows.

Predictor (at the Transmitter):

$$\hat{v}_k(m,n) = \alpha v_{k-1}^*(m,n)$$

$$v_k^*(m,n) = \hat{v}_k(m,n) + e_k^*(m,n)$$

Quantizer:

$$\text{Input: } e_k(m,n) = v_k(m,n) - \hat{v}_k(m,n)$$

$$\text{Output: } e_k^*(m,n)$$

Reconstituted Output at the Receiver:

$$\tilde{v}_k^*(m,n) = \alpha \tilde{v}_{k-1}^* + \tilde{e}_k^*(m,n)$$

where $\tilde{e}_k^*(m,n)$ is the received value of $e_k^*(m,n)$ at the receiver and in the absence of channel noise $\tilde{e}_k^*(m,n) = e_k^*(m,n)$.

For the Markov representation of (5-1), we have [31,86]

$$\sigma_e^2(m,n) \triangleq E[e_k^2(m,n)] = \sigma_v^2(m,n)(1-\alpha^2) , \quad (5-3)$$

where $\sigma_v^2(m,n) \triangleq E[v_k^2(m,n)]$

and the values of $\sigma_v^2(m,n)$ are calculated either from a 2-D spatial domain covariance model or by direct measurement as in the previous chapter.

(See appendix A for details.)

We assume that each $e_k(m,n)$ is Laplacian in distribution and is quantized by its Max quantizer. Let $b_{m,n}$ be the number of bits required to code $e_k(m,n)$. With these assumptions it can be easily shown that (5-3) becomes [75,86]

$$\sigma_e^2(m,n) = \sigma_v^2(m,n) \cdot (1-\alpha^2) / \{1 - q(b_{m,n})\alpha^2\} \quad (5-5)$$

where $q(\cdot)$ is a quantization distortion function defined by (4-5).

Assuming noise-free channel, the average mean square distortion is given by

$$\begin{aligned} D &= \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N \sigma_e^2 q(b_{m,n}) \\ &= \frac{1}{MN} \sum_m \sum_n \sigma_v^2(m,n) \cdot \frac{q(b_{m,n}) (1-\alpha^2)}{1 - q(b_{m,n}) \alpha^2} . \end{aligned} \quad (5-6)$$

As in chapter IV, we assume

$$b_{m,n} = \text{Integer} \geq 0 \quad (5-7)$$

and

$$\sum_{m=1}^M \sum_{n=1}^N b_{m,n} = MNb \quad (5-8)$$

where b is the average bit-rate. The selection of $b_{m,n}$ (or the bit allocation) is made such that (5-6) is minimized. Once again, we use the integer bit allocation algorithm of [86] to achieve this.

5.1 Adaptive Interframe Hybrid Coding Scheme:

The adaptive strategy of section 4.1 can also be applied to the hybrid coding method discussed above with appropriate modification. (In fact this algorithm was first developed for hybrid coding, and then extended to interframe transform coding.)

The activity index of a sub-block is chosen to be its interframe variance (IFV) given by

$$a = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N \{u_k(m,n) - u_{k-1}(m,n)\}^2.$$

Once again, a sub-block is classified into one of four classes by choosing suitable values for the activity index thresholds. Different bit-rates and statistics (prediction coefficient α , and transform coefficient variances $\sigma_v^2(m,n)$) for each class are appropriately selected or measured. This results in a variable bit-rate.

Figure 5-2 shows the schematic of Fig. 5-1 with necessary modifications for the adaptations.

5.2 Hybrid Coding with Motion Compensation

It has been already pointed out in chapter II and elsewhere that the temporal direction (for motion images) primarily consists of a deterministic

component (i.e., motion). It was also shown in chapter II that this component can be satisfactorily modeled by piecewise linear translations, and that motion compensation based on this model results in a tremendous reduction in the interframe variance (hence, improvement in the temporal correlation). Now we consider how the motion measurement methods of chapter II could be incorporated in the hybrid coding schemes discussed above. Let (ℓ_1, ℓ_2) be the motion coordinates of the sub-block U_k relative to the $(k-1)$ th frame. Then the motion compensation is incorporated simply by replacing U_{k-1} by U_{k-1}^C given by

$$U_{k-1}^C(i,j) = u_{k-1}(i+\ell_1, j+\ell_2) \quad 1 \leq i \leq M, 1 \leq j \leq N$$

and thus replacing V_{k-1}^* by

$$V_{k-1}^{C*} = \Psi_M U_{k-1}^{C*} \Psi_M^T.$$

The motion coordinates (ℓ_1, ℓ_2) are coded together with the other information.

The frame skipping and the interpolation of skipped frames can be incorporated in the schemes as described in section 2.5.

5.3 Distortion-Rate Curves from Models of Interframe Motion:

In section 2.1, the relationship between the temporal correlation, the distributions of the interframe motion uncertainty (i.e., \tilde{dx} , \tilde{dy}), and the measurement noise was established. Assuming a first order Markov separable model along the temporal dimension and a model for intraframe covariance, we can thus calculate the distortion-rate functions from the model of motion uncertainty.

Let \tilde{dx} and \tilde{dy} represent the motion uncertainty in pixels/frame along x and y axes, respectively. Let $N(\mu, \sigma)$ and $B(a, b)$ denote the

Gaussian density with mean μ and standard deviation σ , and uniform density in the interval $[a,b]$, respectively. Let us assume that \tilde{dx} and \tilde{dy} are identically distributed. We also assume that images are noise-free, i.e., $\sigma_n^2 = 0$ in (2-6).

For the calculation of distortion-rate functions, we assume the intraframe covariance model to be the isotropic model of (A-2) with $\rho_i = \rho_j = \rho$. We choose two values for parameter ρ , viz., $\rho = .95$ and $\rho = .90$. The first one, which we call Isotropic-1, is a good approximation for the Head and Shoulders images. The second one, which we call Isotropic-2, is a good approximation for the Chemical Plant images.

We have chosen two distributions for \tilde{dx} and \tilde{dy} , the Gaussian and the uniform. For these distributions, we use the approximation of (2-6) to calculate the temporal correlation coefficient, which is used as the prediction coefficient α for the hybrid coding scheme described in this chapter. We choose a sub-block size of 16×16 , and (5-6) for the calculation of distortion.

Figure 5-3 shows some distortion-rate curves for various distributions of \tilde{dx} and \tilde{dy} for unit variance data. The distributions corresponding to the higher variances of \tilde{dx} and \tilde{dy} (i.e., $B(-2,2)$, $B(-4,4)$, $N(0,1)$) could be assumed as reasonable models for coding without motion compensation ($\tilde{dx} = dx$), and those with lower variances (i.e., $B(-.5,.5)$, $N(0,.25)$) for coding with motion compensation. The curve for $\alpha = 0$ corresponds to the intraframe transform coding.

Table 5-1 gives the rates for a fixed distortion for each intraframe covariance model. These are also shown on Fig. 5-3 by dotted lines. We notice that in the absence of motion compensation a hybrid coding scheme achieves a compression gain (over the intraframe transform

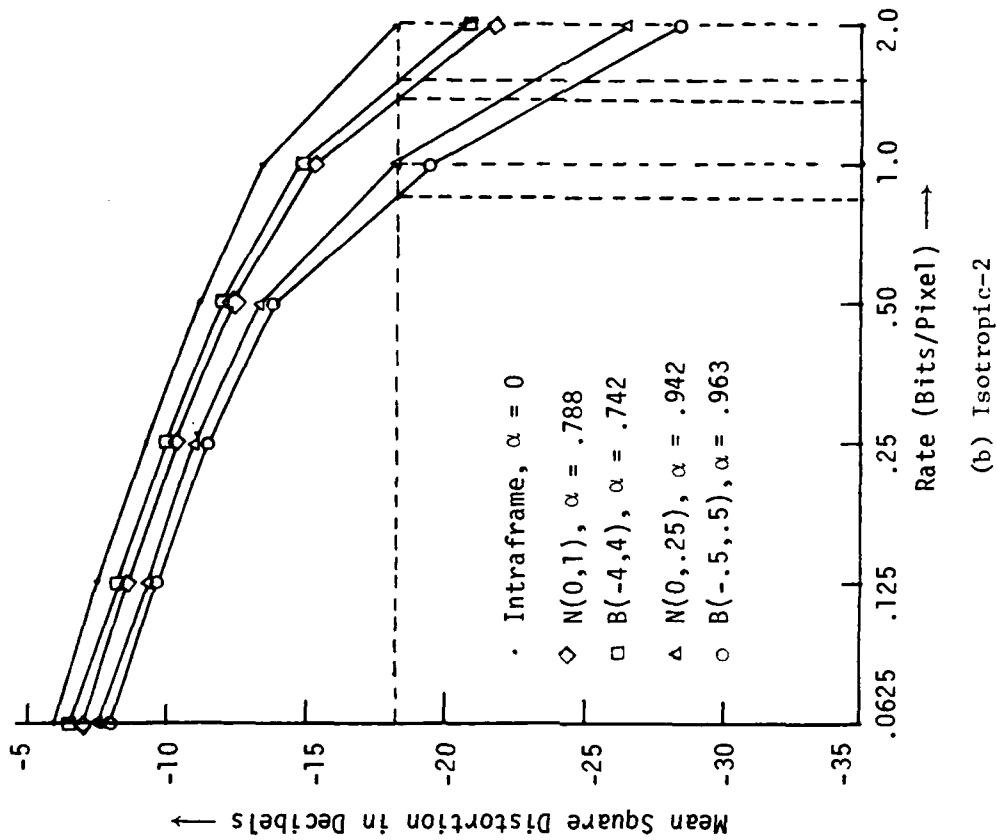
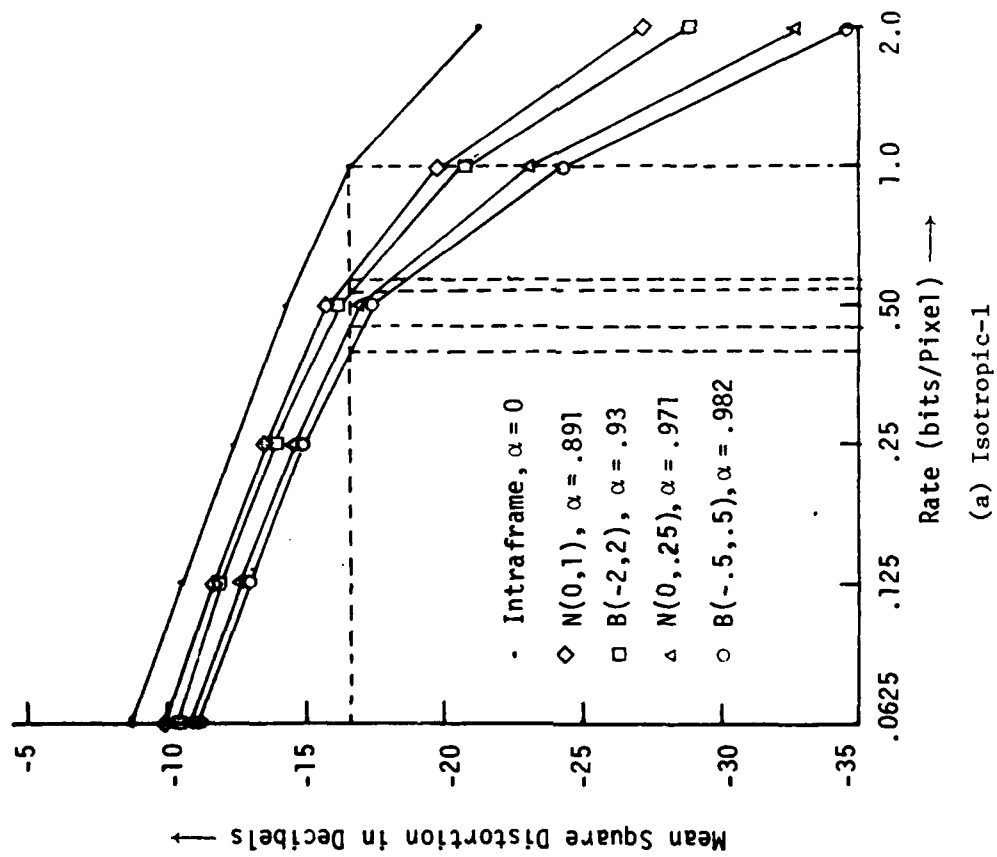


Figure 5-3: Distortion-Rate Curves for Hybrid Interframe Coding of Images with Isotropic Intraframe Covariance and for Various Distributions of Interframe Motion Uncertainty.

Isotropic-2 model, for the distributions considered. Motion compensation results in another additional compression gain (over hybrid coding without motion compensation) by a factor of ≈ 1.35 for Isotropic-1 and ≈ 1.5 for Isotropic-2.

TABLE 5-1
COMPARISONS OF RATES FOR INTERFRAME HYBRID CODING
FOR VARIOUS DISTRIBUTIONS OF MOTION UNCERTAINTY.

DISTRIBUTION OF \tilde{d}_x, \tilde{d}_y	RATE (BITS/PIXEL)	
	ISOTROPIC-1 D = -16.5 DB	ISOTROPIC-2 D = -18 dB
B(-.5,.5)	.4073	.87
N(0,.25)	.4518	1.00
B(-2,2)	.5278	-
B(-4,4)	-	1.49
N(0,1)	.5754	1.34
$\alpha = 0$	1.0	2.0

5.4 Experimental Results:

Since the hybrid coding described above requires knowledge of the initial frame, we assume that the first frame of the data is available as initial condition without any distortion.

5.4.1 Head and Shoulders Images - All the hybrid coding schemes - non-adaptive, adaptive, and adaptive with motion compensation (with and without frame skipping and interpolation) - were applied to this data set. A sub-block size of 16×16 was selected. An isotropic covariance model with correction as described in appendix A has been used for modeling the intraframe statistics.

Table 5-2 gives the performance of the non-adaptive hybrid coding scheme for which the suitable values of the correlation parameters were found to be $\rho_i = .955$, $\rho_j = .945$, $\alpha = .80$. Comparing Table 5-2 with the entries (for the isotropic model with correction) in Table 4-1 we notice an improvement of about 2-2.5 dB at the bit-rates considered, or, equivalently, a compression gain by a factor of about 1.5, as a result of temporal correlation.

Figure 5-4 shows the histogram of the activity index. Although the dynamic range of the activity index is roughly 0-3000, the histogram shows the distribution in the range 0-200 (in which about 90% of the samples are contained) to better utilize the range of the histogram. All the sub-blocks having activity index above 200 have been lumped in the last column. We see that a great number of sub-blocks have a very low activity index (mainly due to stationary background). The threshold values of the activity index chosen are marked by arrows. Our experimental results indicate that the coding performance is not very sensitive to threshold selection.

Tables 5-3 and 5-4 show the parameters and the performance of the 4 class adaptive hybrid scheme. Comparison of the correlation parameters of table 5-3 with those for the non-adaptive hybrid scheme confirms our earlier statement (in chapters II and IV) that the classification based on the activity index (IFV) divides the images into classes of varying spatial activity (characterized by ρ_i and ρ_j) in addition to the varying temporal activity. The improvement due to the adaptations of the bit-rates and the spatial-temporal statistics is about 4-4.5 dB, or equivalently, an additional compression gain (over the non-adaptive hybrid) by a factor of 2.

Figure 5-5 shows the signal to noise ratio as a function of frame number for the non-adaptive and the adaptive hybrid schemes and the intra-

TABLE 5-2

PERFORMANCE OF THE NON-ADAPTIVE HYBRID CODING
SCHEME FOR THE HEAD AND SHOULDERS DATA.

S. N.	BIT-RATE	SIGNAL TO NOISE RATIO
1	.25	32.28 dB
2	.50	35.49 dB

TABLE 5-3

PARAMETERS OF THE 4 CLASS ADAPTIVE HYBRID
SCHEME FOR HEAD AND SHOULDERS DATA.

CLASS NO.	ACTIVITY INDEX	PROBABILITY OF OCCURENCE	CORRELATION PARAMETERS		
			ρ_i	ρ_j	α
1	0-20	.5768	.98	.975	.98
2	20-60	.1737	.945	.94	.93
3	60-200	.1320	.92	.905	.80
4	200-	.1174	.86	.84	.40

TABLE 5-4

PERFORMANCE OF THE ADAPTIVE HYBRID CODING SCHEME
FOR THE HEAD AND SHOULDERS DATA.

S. N.	BIT-RATE					SIGNAL TO NOISE RATIO IN DECIBELS				
	CLASS #				OVER- ALL	CLASS #				OVER- ALL
	1	2	3	4		1	2	3	4	
1	.03	.10	.18	.40	.114	34.82	31.85	31.01	30.32	32.86
2	.075	.30	.45	.75	.25	37.35	35.95	35.65	35.03	36.52
3	.18	.70	.90	1.25	.50	39.80	39.89	40.23	39.70	39.85

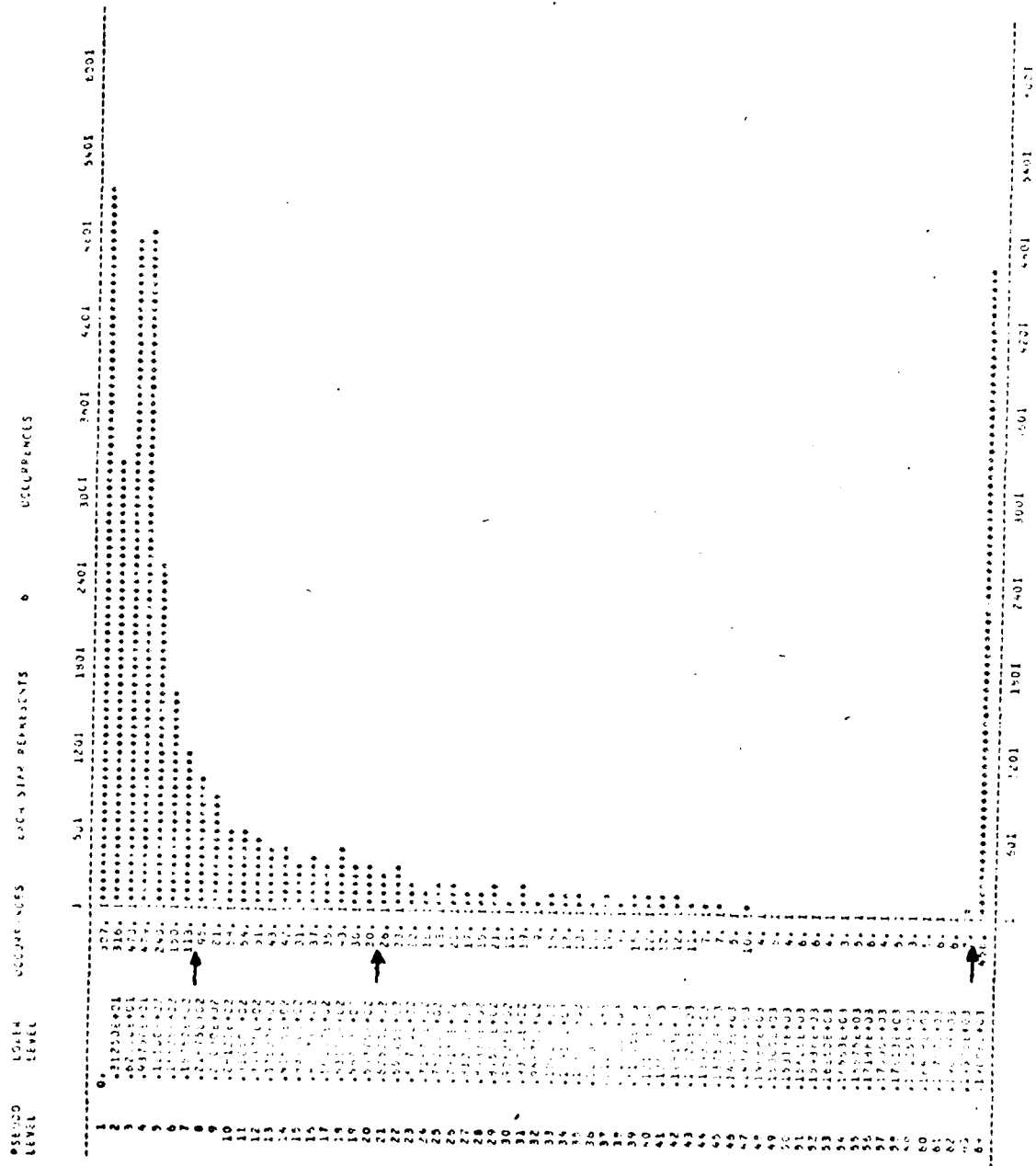


Figure 5-4: Histogram of the Activity Index (IFV) of Head and Shoulders Images. Sub-Block Size = 16×16 .

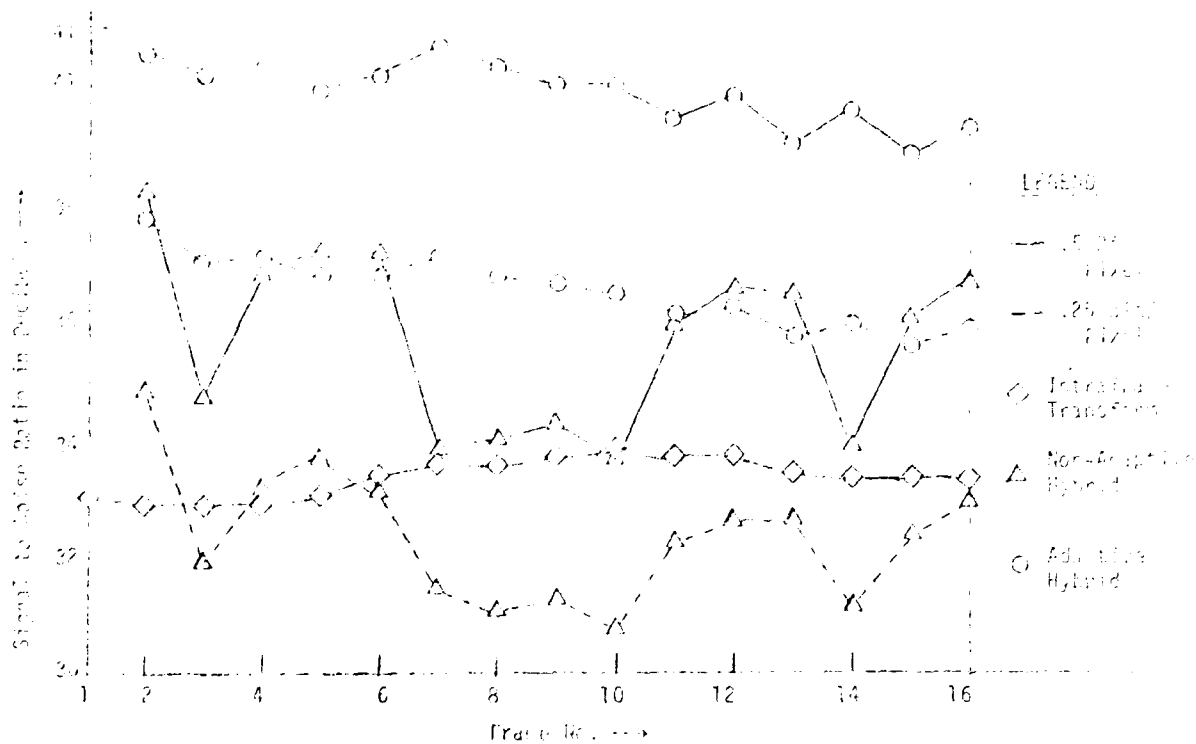


Figure 5-5: Signal to Noise Ratio as a Function of Frame Number for Coding of the Head and Shoulders Images.

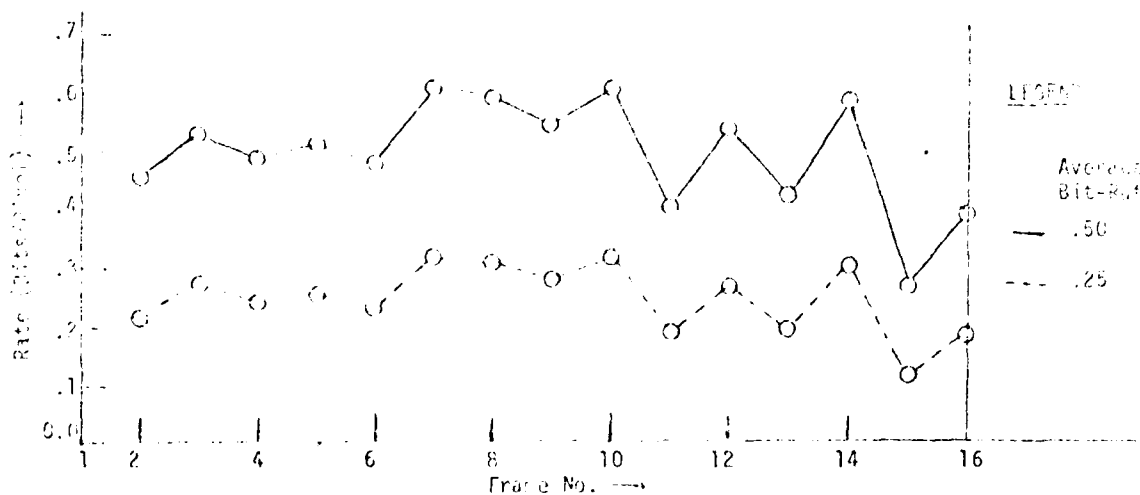


Figure 5-6: Bit-Rate as a Function of Frame Number for Adaptive Hybrid Coding of the Head and Shoulders Images.

frame transform coding scheme (which is a special case of the non-adaptive hybrid scheme when $\alpha = 0$). We make the following observations and comments from the figure. (i) The SNR is almost constant for the intraframe transform scheme. This is expected because the spatial contents of the frames are very similar. (ii) The SNR varies greatly as a function of frame number for the non-adaptive hybrid coding scheme. This variation is primarily due to the variation in the amount of motion activity. The frames having larger motion (e.g., 7 through 10) have low SNR, which is very close to the SNR of the intraframe scheme. (iii) For the adaptive hybrid scheme, the SNR is fairly constant (as would be expected from a good adaptive scheme). Thus, the adaptations proposed are effective.

Figure 5-6 shows the bit-rate as a function of frame number for the adaptive hybrid coding. The frames containing larger motion have higher bit-rates (as would be expected). However, the bit-rate variation is not as rapid as would be expected from a predictive coding scheme such as the frame replenishment cluster coding described in section 3.1 (if the rate is not controlled by the buffer fullness).

Figure 5-7 shows some of the images of frame #8 resulting from adaptive and non-adaptive hybrid coding. Comparing the images of Figs. 5-7(a) (non-adaptive hybrid) and 4-2(a)-(i) (intraframe transform), we notice that the non-adaptive hybrid scheme reproduces moving areas very poorly as compared with the intraframe transform scheme. However, the stationary areas are better reproduced. Note that the only difference between the coding of these two images is the temporal prediction coefficient (which is .8 for the hybrid scheme and 0.0 for the intraframe transform scheme).



(a) Non-adaptive, Bit-rate = .5 bit/pixel,
SNR = 34.00 dB.



(b) Adaptive, Bit-rate = .114 bit/pixel,
SNR = 33.04 dB.



(c) Adaptive, Bit-rate = .25 bit/pixel,
SNR = 36.78 dB.



(d) Adaptive, Bit-rate = .5 bit/pixel,
SNR = 40.25 dB.

Interframe Hybrid Coding using Isotropic Model with Correction, Sub-block Size = 16 x 16.

Figure 5-7

At low bit-rates (Figure 5-7(b)) we notice a shadow (or ghost) near the sharp edges of moving areas for the adaptive hybrid scheme. If a smoothing filter is employed in the temporal direction, this distortion will change to blurring of the moving areas (the interframe transform coding does that, as can be seen from Figure 4-4(c)). At higher bit-rates the shadow effect diminishes and the visual quality of the adaptive hybrid and adaptive interframe transform schemes are comparable. Thus from complexity consideration, the adaptive hybrid coding scheme is more attractive.

On comparing the adaptive hybrid coding scheme presented here with the one presented in [67] we note the following - (i) the mean square performance is slightly (1-2 dB) better for the scheme presented here at higher bit-rates ($\geq .5$); (ii) the scheme presented in [67] is computationally more complex; (iii) the scheme presented in [67] has a fixed bit-rate (overall as well as for each sub-block), while the one presented here has a variable bit-rate; (iv) the scheme presented here reproduces moving edges more accurately (because of the higher bit-rate in the areas containing moving edges).

Now we present the results for the adaptive hybrid coding with motion compensation. In these experiments only frames 5 through 9 are used (to minimize computational costs). However, we have no reason to believe that the results will be significantly different if all the 16 frames were employed. This is because, with motion compensation, the interframe activity (measured by IFV) is almost a constant function of frame number (as was shown in section 2.4). Thus, the distortion and/or the frame bit-rate would now be (as opposed to the case of no motion compensation) independent of frame number. Also, as is evident from

Figure 5-5, the adaptive hybrid coding scheme achieves the steady state very fast (right after the first coded frame). For results on motion measurements, frame skipping, interpolation, etc., for these frames, see section 2.4.

Tables 5-5 and 5-6 give the parameters of the adaptive hybrid scheme with motion compensation without and with alternate frame skipping, respectively. We note the following - (i) Due to change in the distribution of the activity index (greatly reduced), as a result of motion compensation, the thresholds for classification have been lowered; (ii) There is a great improvement in the temporal correlation as evidenced by the values of α . We would also like to point out that the average temporal activity (measured by average motion in pixels/frame of a class is not directly evident from the value of α which is very nearly the temporal correlation parameter). Let h represent average temporal activity after motion compensation. Let $\rho_i = \rho_j = \rho$. Let the intraframe covariance be given by the isotropic model. Then, an approximate value of h is given by

$$\begin{aligned}\alpha &\approx \rho^h \\ \text{or } h &\approx \frac{\ln(\alpha)}{\ln(\rho)} \\ &\approx \frac{\ln\{1-(1-\alpha)\}}{\ln\{1-(1-\rho)\}} \\ h &\approx \frac{1-\alpha}{1-\rho} .\end{aligned}$$

Computing this quantity from the entries of Table 5-5 we note that it is very close to .25 for all the four classes, which further supports our conclusions in chapter II that the motion uncertainty, after motion compensation, is fairly uniformly distributed over various classes. Thus, the

TABLE 5-5

PARAMETERS OF THE 4 CLASS ADAPTIVE HYBRID CODING SCHEME
WITH MOTION COMPENSATION FOR HEAD AND SHOULDERS DATA.

CLASS NO.	ACTIVITY INDEX	PROBABILITY OF OCCURRENCE	CORRELATION PARAMETERS		
			ρ_i	ρ_i	α
1	0-10	.506	.985	.98	.996
2	10-20	.295	.955	.945	.99
3	20-50	.143	.91	.90	.97
4	50-	.057	.80	.78	.95

TABLE 5-6

PARAMETERS OF THE 4 CLASS ADAPTIVE HYBRID CODING SCHEME WITH
MOTION COMPENSATION, USING ALTERNATE FRAME SKIPPING AND INTER-
POLATION, FOR HEAD AND SHOULDERS DATA.

CLASS NO.	ACTIVITY INDEX	PROBABILITY OF OCCURRENCE	CORRELATION PARAMETERS		
			ρ_i	ρ_i	α
1	0-20	.67	.98	.975	.99
2	20-40	.184	.95	.94	.97
3	40-100	.102	.91	.90	.95
4	100-	.045	.80	.78	.90

TABLE 5-7

PERFORMANCE OF THE ADAPTIVE HYBRID CODING WITH MOTION COMPENSATION
FOR HEAD AND SHOULDERS FRAMES 5 THRU 9. SKIPPED FRAMES ARE INTER-
POLATED ALONG THE MOTION TRAJECTORY.

FRAMES SKIPPED?	AVERAGE BIT-RATE PER PIXEL	SIGNAL TO NOISE RATIO IN DECIBELS		
		CODED FRAMES	INTERPOLATED FRAMES	OVERALL
NO	.253	38.74	-	38.74
YES	.252	39.97	37.58	38.62
YES	.125	37.60	36.69	37.12

adaptation due to classification in this scheme is used to compensate for the spatial non-stationarity. If the images are spatially stationary, this scheme will reduce to the non-adaptive hybrid coding scheme with motion compensation.

Table 5-7 gives the coding results for the adaptive hybrid scheme with motion compensation with and without frame skipping. At an SNR of about 37 dB a compression gain by a factor of two can be achieved over the adaptive scheme without motion compensation. Figure 5-8 (parts (b)-(d)) show the resulting images corresponding to frame 8. Visual quality of these images is very good. The results indicate that at very low bit-rates (.125 and below) the adaptive hybrid coding scheme with motion compensation and frame skipping, and interpolation of skipped frames along the motion trajectory, is very promising for high quality fidelity encoding of motion images.

In the experiments on coding with motion compensation, the direction of minimum distortion (DMD) method described in section 2.3 was used with mean square distortion criterion for the measurement of motion.

5.4.2 Chemical Plant Images: Due to very low spatial correlation and large temporal activity, the hybrid schemes without motion compensation (non-adaptive and adaptive) are expected to result in no significant improvement over an intraframe transform coding scheme. Therefore, the adaptive hybrid coding scheme with motion compensation was used. A sub-block size of 16×16 was selected.

The motion measurement was done by the area correlation method with a Fourier domain filtering given by (2-12) (see section 2.3 for details). The value of $\gamma = .5$ was found to yield good results. For this data set,



(a) No Frame Skipping, Bit-rate = .5 bit/pixel,
SNR = 30.45 dB.



(b) No Frame Skipping, Bit-rate = .25 bit/pixel,
SNR = 38.51 dB.



(i) Bit-rate = .125 bit/pixel, SNR = 36.70 dB.



(ii) Bit-rate = .25 bit/pixel, SNR = 37.59 dB.

(c) Alternate Frame Skipping, Interpolated Image along Motion Trajectory.

Interframe Adaptive Hybrid Coding with Motion Compensation using Measured Statistics.

Figure 5-8



the performance of the area correlation method and the DMD method of motion measurement is comparable (with DMD being slightly superior).

The intraframe statistics were measured (in the transform domain) for each of the classes. Tables 5-8 and 5-9 give the parameters and the performance of the coding scheme. Figure 5-8(a) shows an image corresponding to frame 12. Comparing these results with those of the intraframe transform scheme with the same sub-block size, we note an improvement of about 2.5-3 dB. The compression gain (over the intraframe scheme) at a distortion level of 30 dB is by a factor of about 2, and smaller at higher distortion levels.

Thus, we see that the gains due to adaptation and motion compensation are much lower than those for the Head and Shoulders images. From the results of section 2.5, it is evident that the frame skipping and interpolation along motion trajectory can be successfully used for these images to achieve higher compression similar to those for the Head and Shoulders images.

5.4.3 Angiocardiogram Images - The temporal activity of these images exhibit two characteristics of the cardiac cycle - (i) it is periodic; (ii) it is nonuniformly distributed over a period (cardiac cycle). Also, at the frame sampling rate of 1/30 sec., the images have high temporal correlation.

Due to the above properties, the adaptive hybrid coding scheme (without motion compensation) was found to be ideally suited for these images. The images were found to be spatially stationary. This was expected, because these images do not exhibit sharp features which are characteristic of most video images. Therefore, adapting the intraframe statistics to classes of different activity index is not necessary.

TABLE 5-8

PARAMETERS OF THE 4 CLASS ADAPTIVE HYBRID CODING WITH
MOTION COMPENSATION FOR THE CHEMICAL PLANT IMAGES.

CLASS #	ACTIVITY INDEX	PROBABILITY OF OCCURRENCE	TEMPORAL PREDICTION COEFFICIENT α
1	0-100	.4773	.90
2	100-300	.4143	.75
3	300-750	.0956	.60
4	750-	.0128	.40

TABLE 5-9

PERFORMANCE OF THE ADAPTIVE HYBRID CODING WITH MOTION COMPENSATION
FOR THE CHEMICAL PLANT IMAGES. SUB-BLOCK SIZE = 16×16 .

S. N.	BIT-RATE					SIGNAL TO NOISE RATIO IN DECIBELS				
	CLASS #				OVER- ALL	CLASS #				OVER- ALL
	1	2	3	4		1	2	3	4	
1	.148	.70	1.0	.875	.5	30.31	30.22	30.09	30.59	30.25
2	.55	1.3	1.5	1.5	1.0	33.77	34.06	33.34	34.59	33.85
3	1.75	2.15	2.25	2.25	2.0	39.57	38.71	37.67	38.85	38.98

Figure 5-9 shows the bit allocation at two bit-rates. Bit allocation pattern for these images is very unusual compared with video images. The data seems to have some characteristic frequencies (more accurately speaking, the discrete Cosine transform basis vectors). An attempt to model the statistics by any of the commonly used models would result in a loss of these frequencies, and thereby, a probable loss of medically useful information for the same signal to noise ratio.

Tables 5-10 and 5-11 give the parameters and the performance of the coding scheme. Figure 5-10 shows the classification maps for two of the frames. It is interesting to note from Fig. 5-10 that the classification scheme very closely follows the activity of the cardiac cycle. During a stationary cardiac frame period, the scheme uses less than one-third the average bits/frame and during an active period, about twice the average rate.

Figure 5-11 shows two of the original frames (combined into a single image) and their coded equivalents at some of the compression ratios. Even at very low bit-rate of .0625 (or a compression ratio of 128) the image quality looks fair (by evaluation of still frames). For these images a compression ratio of 32 to 128 seems to be realizable. An accurate reproduction of motion is required for these images. The methods using the exchange of spatial and temporal resolution, which are acceptable for the video images, could not be used for these images.

TABLE 5-10
PARAMETERS OF A 4 CLASS ADAPTIVE HYBRID CODING SCHEME FOR
THE ANGIOCARDIOGRAM IMAGES.

CLASS #	ACTIVITY INDEX	PROBABILITY OF OCCURRENCE	TEMPORAL CORRELATION α
1	0. - 10.	.570	.98
2	10. - 25.	.282	.90
3	25. - 60.	.103	.85
4	60. - 00	.045	.75

TABLE 5-11
PERFORMANCE OF THE ADAPTIVE HYBRID CODING SCHEME FOR THE ANGIOCARDIOGRAM
IMAGES. SAME MEASURED STATISTICS WERE USED FOR ALL CLASSES. SUB-BLOCK
SIZE = 16×16 .

S. N.	BIT-RATE PER PIXEL					COMPRESSION RATIO	SIGNAL TO NOISE RATIO IN dB				
	CLASS 1	CLASS 2	CLASS 3	CLASS 4	OVER ALL		CLASS 1	CLASS 2	CLASS 3	CLASS 4	OVER ALL
1	.02	.05	.13	.35	.0625	128	35.95	34.34	33.44	33.37	35.00
2	.039	.12	.28	.70	.1250	64	38.39	37.24	36.19	36.17	37.66
3	.10	.25	.60	1.17	.250	32	40.61	39.35	38.81	38.02	39.87
4	.25	.60	1.0	1.7	.50	16	42.08	41.59	40.73	39.67	41.64



(a) Original



(b) C.R. = 128



(c) C.R. = 32



(d) C.R. = 16

Images resulting from data compression of angiogram images. The top half of each image approximately corresponds to an end of systole and the bottom half to the end of the following diastole.

Figure 5-11

CHAPTER VI

DATA COMPRESSION FOR NOISY CHANNELS

In the data compression designs considered in the previous chapters we did not consider the effects of channel errors (in transmission or storage-retrieval). The performance of a data compression method was evaluated assuming a noise-free channel. However, in the presence of channel errors (bit reversals) a coding scheme designed without regard to channel noise characteristics could yield poor to disastrous results.

A common approach for reducing the effects of channel errors has been the use of error correcting codes [45] which aim at minimizing the probability of bit error by introducing redundancy in the code word (blocks 4 and 6 of Fig. 1-1). However, a better design would be to incorporate channel characteristics in the data compression algorithm itself, (blocks 3 and 7 in Fig. 1-1), e.g., in the design of quantizer [41], design of predictor coefficients for DPCM transmission [14], periodic reinitialization of DPCM loops, etc.

Most conventional error correcting codes provide equal protection to all the bits for a Gaussian binary symmetric channel. Often, all the data bits do not have equal importance. For example, in transform coding the transform coefficients have highly uneven distribution of mean square energy and different bits of the same coefficient have unequal effect on the mean square energy. For example, in a transform image coding scheme described in [51] certain bits which have "significant" effects on image quality are identified and only these bits are provided protection by using error correcting codes. However, the experimental method used

there for identifying the significant bits is tedious and does not have any systematic quantitative formulation. Also, all the significant bits are provided equal protection, although their effects on the image quality vary considerably.

6.1 Channel Encoding-Decoding of a Random Variable with MSE Criterion:

Crimmins, Horwitz, et al. [18,19] have proposed an alternative method of encoding numerical data. Their method is based on minimizing the mean square error (MSE) due to channel noise rather than minimizing the probability of bit-error. They find the optimum encoding and decoding rules for transmitting a set S of equispaced and equiprobable real numbers over a memoryless channel using certain group (or block) codes. The set S contains K elements where $K = 2^k$, integer $k \geq 0$, and the code words are chosen from a given group G of order K . Each member of group G is a code word of length n bits ($n \geq k$). G is thus a subgroup of the binary group V containing all the code words of length n . Both G and V are groups under exclusive-or operation (denoted by \oplus).

In [18,19] the decoding rule is restricted to map back into the set S . Wolf and Redinbo [78,79] have extended these results to the case where the optimum decoder maps into the field R of real numbers. No method, other than exhaustive search, for finding the optimum subgroup G has been found.

Usually the finite set S contains the quantized values of a continuous random variable, say y . For example if y is a random variable representing the intensity of an image, then $S = \{0,1,\dots,255\}$ could represent digitized values of pixels for 8 bits/pixel digitization.

Except when y is uniformly distributed the set S of its quantized values cannot be equispaced and equiprobable at the same time for $k > 1$. If y is non-uniformly distributed and is quantized using the minimum mean square error Max quantizer, the set S is neither equispaced nor equiprobable for $k > 1$. The procedure of [79] for finding the optimum encoding rule does not apply to such cases. This still remains an open problem. The optimum decoding rule is still given by the conditional mean [79]. Based on our experimental results we believe that the codes generated by the method of [79] could be used for the nonuniformly distributed y with great advantage over the conventional error correcting codes.

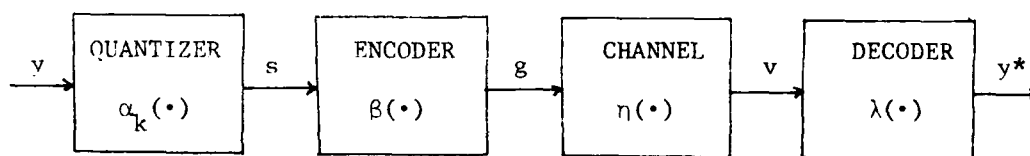


Figure 6-1: PCM Transmission of a continuous Random Variable having zero mean and unit variance over a noisy channel

Figure 6-1 shows a PCM transmission scheme for noisy channels. Let α_k be a k -bit quantization function of a random variable y defined by

$$\alpha_k(y) = s_j \quad \text{for } y_j \leq y \leq y_{j+1}, \quad y_j \leq s_j \leq y_{j+1},$$

$$j = 1, 2, \dots, K$$

where y_1 and y_{K+1} are the minimum and the maximum values of the random variable y and

$$y_i < y_j \quad \text{if } i < j.$$

Then it follows that $S = \{s_1, s_2, \dots, s_K\}$ is an ascendingly ordered set of real numbers. Let H be the group of all k -bit binary code words arranged in the natural order, i.e., the j th element of H is the k -bit binary representation of the integer j . Let

$$\mu_k : S \xrightarrow{1-1} H$$

be an ordered mapping of S onto H . We would like to point out that under this mapping the optimum encoding procedure of [79] provides maximum protection to the most significant bit of $h \in H$ and increasingly lesser protection to the lesser significant bits. For most distributions and quantizers of practical interest this results in a significantly better mean square performance than equal bit error protection encoding. This is our rationale behind using the mean square encoding procedure of [79] even though it is not optimum for nonuniform distributions. Let

$$\begin{aligned} \theta_{n,k} : H &\xrightarrow{1-1} G \\ \beta : S &\xrightarrow{1-1} G, \end{aligned}$$

then $\beta = \theta_{n,k} \odot \mu_k$ is the encoding rule, where \odot represents composite function operation.

The channel error function μ transforms an n -bit code word $g \in G$ into another n -bit word $v \in V$ randomly and is described by the transition probability $P(v|g)$. Let

$$v = g \oplus u$$

where u is an n bit error word. Let us assume that the channel is memoryless, i.e.,

$$P(v|g) = P(v \oplus g|0) = P(u|0)$$

where 0 is the identity element of V . The function η is then completely characterized by $P(u|0)$, $\forall u \in V$. The decoding function λ maps the n -bit word v into a real number y^* . We define

$$\begin{aligned} e_q &= s-y, & q(k) &= [e_q^2]/\sigma^2 \\ e_c &= y^*-s, & c(n,k) &= E[e_c^2]/\sigma^2 \\ e_t &= y^*-y, & t(n,k) &= E[e_t^2]/\sigma^2 \end{aligned}$$

where q , c , and t are the quantization, the channel, and the total (quantization plus channel) mean square distortion functions and σ^2 is the variance of y . The optimization problem can now be stated as follows: Given n, η and the distribution of y , find $k, \alpha, \beta, \lambda$ and G such that the total distortion, $t(n,k)$ is minimized.

The problem as stated above is quite difficult and the joint optimization of the quantization and the encoding seems untractable at the present. A solution for a special case of the above problem has been given by Kurtenbach and Wintz [41]. They assume $k = n$ (which implies $G \equiv V$), fixed β (e.g., $\beta = \mu$), and $\lambda = \beta^{-1}$ and find an optimum quantizer α . This does not provide protection for channel errors by introducing redundancy. The performance of such a scheme is usually not as good as those which do provide protection by introducing redundancy.

To simplify the problem, we separate the quantization and the encoding. We choose α to be the optimum Max quantization function [47], which minimizes the quantization distortion q . For any given β and G the decoder λ which minimizes the channel distortion c is given by the conditional mean [78,79], i.e.,

$$\lambda(v) = E[s|v] \quad (6-1)$$

$$\begin{aligned}
 &= \sum_{s \in S} s \cdot P(s|v) \\
 &= \sum_{s \in S} s \cdot P(v|s) \cdot P(s) / P(v) \\
 &= \sum_{s \in S} s \cdot P(v|\beta(s)) \cdot P(s) / P(v) \\
 &= \frac{\sum_{s \in S} s \cdot P(v \oplus \beta(s)|0) \cdot P(s)}{\sum_{s \in S} P(s) \cdot P(v \oplus \beta(s)|0)} \quad (6-2)
 \end{aligned}$$

where $P(s)$ is the probability of the quantizer output s , which can be calculated for any given α from the distribution of y . If α is the Max quantization function, then it can be verified that (6-1) also implies $\lambda = E[y|v]$. Let $P(v,s)$ be the joint distribution of the random variables s and v . Then the channel distortion is given by

$$\begin{aligned}
 c_{n,k} &= E[\{y^* - s\}^2] \\
 &= E[\{\lambda(v) - s\}^2] \\
 &= \sum_{s \in S} \sum_{v \in V} \{\lambda(v) - s\}^2 P(v,s) \\
 &= \sum_{s \in S} \sum_{v \in V} \{\lambda(v) - s\}^2 P(v|s) P(s) \\
 &= \sum_{s \in S} \sum_{v \in V} \{\lambda(v) - s\}^2 P(v \oplus \beta(s)|0) \cdot P(s). \quad (6-3)
 \end{aligned}$$

Now we establish the following relationships between y , s , y^* , e_q , e_c and e_t .

Theorem 6-1: If α is the Max quantization function and λ is given by (6-1), then the following hold

$$(i) E[s \cdot e_q] = 0 \quad (6-4)$$

$$(ii) E[y^* \cdot e_c] = 0 \quad (6-5)$$

$$(iii) E[y^* \cdot e_t] = 0 \quad (6-6)$$

Proof: Let $f_Y(y)$ be the probability density function of y . Then the following holds true for the Max quantization [47]

$$s_i = \int_{y_i}^{y_{i+1}} y \cdot f_Y(y) dy \quad 1 \leq i \leq K. \quad (6-7)$$

Part (i) of the theorem is a well known result for the Max quantizer and will not be proved here.

The right hand side of (6-5) can be written as

$$\begin{aligned} E[y^* \cdot e_c] &= E[y^*(y^* - s)] \\ &= E[y^{*2}] - E[y^* s]. \end{aligned} \quad (6-8)$$

Now

$$\begin{aligned} E[y^* s] &= E[\lambda(v) \cdot s] \\ &= \sum_{s \in S} \sum_{v \in V} s \cdot \lambda(v) \cdot P(s, v) \\ &= \sum_s \sum_v s \cdot \lambda(v) \cdot P(s|v) P(v) \\ &= \sum_v \{ \lambda(v) \cdot P(v) \sum_s s \cdot P(s|v) \} \\ &= \sum_v \{ \lambda(v) \cdot P(v) \cdot E[s|v] \} \end{aligned}$$

$$\begin{aligned}
&= \sum_v \lambda(v) \cdot P(v) \cdot \lambda(v) \quad [\text{using (6-1)}] \\
&= E[\{\lambda(v)\}^2] \\
&= E[y^{*2}] \quad (6-9)
\end{aligned}$$

Thus, from (6-8) and (6-9) we have

$$E[y^* \cdot e_c] = 0$$

which proves (6-5).

The left hand side of (6-6) can be written as

$$\begin{aligned}
E[y^* e_t] &= E[y^*(y^* - y)] \\
&= E[y^{*2}] - E[y^* y]. \quad (6-10)
\end{aligned}$$

Since

$$\begin{aligned}
E[y^* y] &= \sum_{v \in V} \lambda(v) \cdot \int_{-\infty}^{\infty} y f_{Y|v}(y) dy \cdot P(v) \\
&= \sum_v \lambda(v) \cdot P(v) \cdot \left\{ \sum_{i=1}^K \int_{y_i}^{y_{i+1}} y f_Y(y) dy \cdot P(s_i | v) \right\}.
\end{aligned}$$

Using (6-7) we have

$$\begin{aligned}
E[y^*y] &= \sum_v \lambda(v) \cdot P(v) \cdot \left\{ \sum_{i=1}^K s_i P(s_i|v) \right\} \\
&= \sum_v \lambda(v) \cdot P(v) \cdot E[s|v] \\
&= \sum_v \lambda(v) \cdot P(v) \cdot \lambda(v) \\
&= E[\{\lambda(v)\}^2] \\
&= E[y^{*2}] .
\end{aligned} \tag{6-11}$$

Thus, from (6-10) and (6-11) we have

$$E[y^* \cdot e_t] = 0$$

which proves (6-6) and completes the proof of the Theorem 6-1.

Corollary: The errors due to the quantization and the channel noise are uncorrelated, i.e.,

$$E[e_q e_c] = 0 . \tag{6-12}$$

Proof:

$$\begin{aligned}
E[e_q e_c] &= E[e_q (y^* - s)] \\
&= E[e_q y^*] - E[e_q s] \\
&= E[y^* (e_t - e_c)] - E[e_q s] \\
&= E[y^* e_t] - E[y^* e_c] - E[s e_q] \\
&= 0
\end{aligned}$$

which follows from the theorem. A direct consequence of (6-12) is

$$t(n,k) = c(n,k) + q(k) . \tag{6-13}$$

For a given n , the optimum value of k , k' , is found by computing $t(n,k)$ for $0 \leq k \leq n$ and finding the minima of k vs $t(n,k)$. Let

$$d(n) = \min_k \{t(n,k)\} = t(n,k'). \quad (6-14)$$

Then $d(n)$ vs n gives the distortion-rate function of the PCM channel. We call the optimization of (6-14) channel optimization.

For uniform distribution of y the optimum encoding function β for a given G can be found as in [79]. The same β could be used for some other distributions of practical interest as pointed out earlier. No simple method for finding the optimum β for such cases has been found so far.

6.2 Coding of a Random Process for Noisy Channels:

The concept of channel optimization for a single random variable could be extended for coding images and other correlated signals or random processes. In particular we consider the transform coding method. With respect to channel errors transform coding has an advantage over the predictive or hybrid methods. Since each transform coefficient is coded completely independently, any error due to the channel noise does not effect the other coefficients. On the other hand in predictive coding, the errors due to channel noise accumulate at the time of reconstruction at the receiver. This is because unlike the quantization errors the channel errors cannot be fed back in the prediction loop at the transmitter. For optimization of prediction loops for DPCM transmission of images over noisy channels, see [14].

Figure 6-2 shows a transform coding scheme with channel optimization. X is an $M \times 1$ real array and

$$y = \Psi x$$

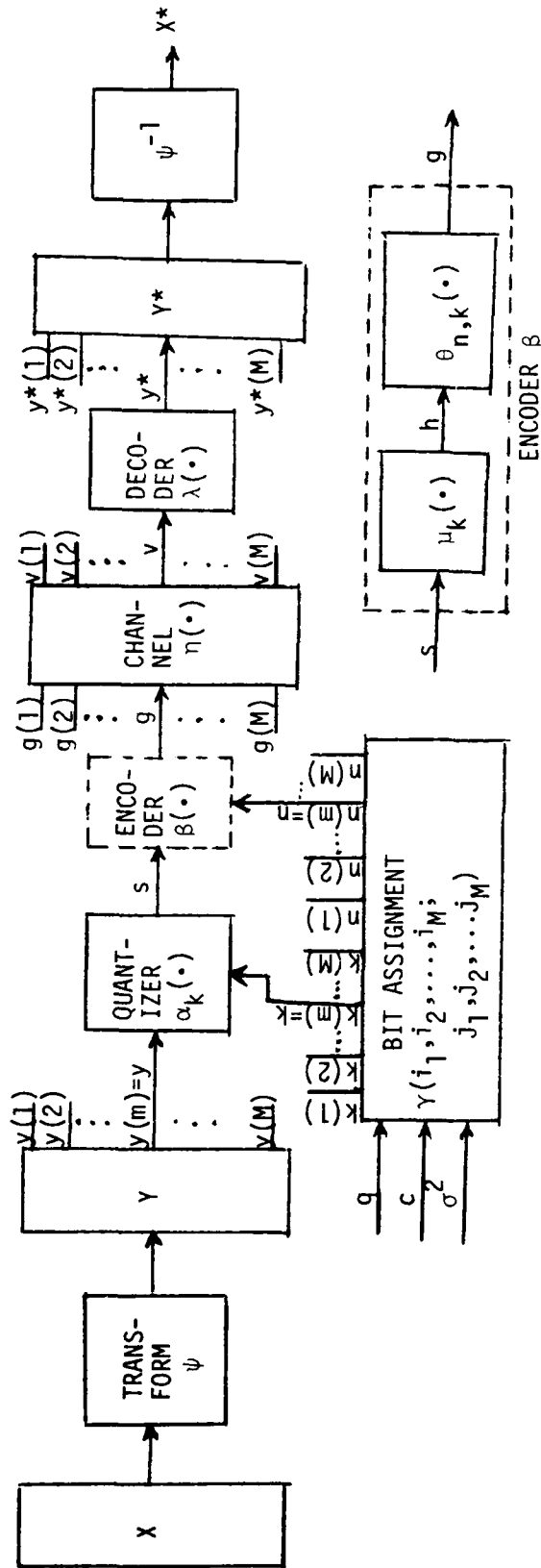


Figure 6-2: A Data Compression Scheme for Noisy Channels.

is an $M \times 1$ array of transform coefficients, where Ψ is an $M \times M$ unitary transform. We assume for simplicity that all the transform coefficients are real, identical distributed, and have zero mean. Let

$$\sigma^2(m) \triangleq E[y^2(m)]$$

where $y(m)$ is the m th transform coefficient. Let us also assume that the columns of the transform matrix Ψ are arranged in an order such that

$$\sigma^2(1) \geq \sigma^2(2) \geq \dots \geq \sigma^2(M). \quad (6-15)$$

Then the total distortion between the input array X and the reproduced array X^* is given by

$$\begin{aligned} D &= E[(X-X^*)^T(X-X^*)] \\ &= E[(Y-Y^*)^T(Y-Y^*)] \\ &= \sum_{m=1}^M E[\{y(m) - y^*(m)\}^2] \\ &= \sum_{m=1}^M \sigma^2(m) \cdot t(n(m), k(m)) . \end{aligned}$$

Assuming that for each $n(m)$ the optimum value $k'(m)$ is used, we obtain

$$D = \sum_{m=1}^M \sigma^2(m) \cdot d(n(m)) . \quad (6-16)$$

We would like to minimize D subject to the constraint

$$\frac{1}{M} \sum_{m=1}^M n(m) = b \quad (6-17)$$

where b is a given rate in bits/sample. We assume that β and G for various values of pair (n, k) are chosen such that the distortion-rate

function $d(n)$ is a non-increasing function of n . This condition can always be satisfied in practice. The minimization of (6-16) subject to (6-17) is identical to the bit allocation problem discussed in chapter IV except that the quantizer distortion function, q , has been replaced by the optimum total distortion function, d . However, it is not easy to approximate d by piecewise continuous functions as has been done for the quantizer distortion, q , for some commonly used densities [86]. Thus, the use of integer bit allocation procedure of [86] becomes even more important in this case.

It could be easily verified that the K-L transform would be the optimum unitary transform for the scheme described above.

6.3 Experimental Results and Distortion-Rate Functions:

We have carried out simulations for an important class of channels, the binary symmetric channel with probability of bit-reversal p . We report results for the PCM transmission of a random variable (without loss of generality we assume zero mean and unit variance) for three commonly used densities, the Gaussian, the Laplacian, and the uniform. The uniform density also gives the lower bounds for the quantizer distortion, $q(k)$, and the channel distortion normalized by the variance of s , i.e., $c(n,k)/(1-q(k))$. For quantization we use an approximation of the optimum (Max) quantizer described in [4]. This approximation is very close to the optimum. Table 6-1 gives the distortion-rate functions for the three densities.

Using the algorithm outlined in [78], we have found some suitable choices of the group G for various values of the pair (n,k) . Table 6-2

TABLE 6-1

QUANTIZER DISTORTION $q(k)$ FOR VARIOUS DENSITIES.

Density k	1	2	3	4	5	6	7	8
UNIFORM	.25	.0625	.0156	.0039	.00098	2.4×10^{-4}	6×10^{-5}	1.5×10^{-5}
GAUSSIAN	.368	.1204	.0356	.0098	.0026	.0006	.00017	4×10^{-5}
LAPLACIAN	.5001	.1835	.0571	.0160	.0042	.0011	.00027	7×10^{-5}

TABLE 6-2

BASIS VECTORS $\{g_i; i=1, \dots, k\}$ OF GROUP G FOR (n, k) GROUP CODES.

i	k \ n-k	0	1	2	3	4
1	8	10000000	100000001	1000000011	10000000110	
2		01000000	010000000	0100000001	01000000101	
3		00100000	001000000	0010000000	00100000011	
4		00010000	000100000	0001000000	00010000111	
5		00001000	000010000	0000100000	00001000000	
6		00000100	000001000	0000010000	00000100000	
7		00000010	000000100	0000001000	00000010000	
8		00000001	000000010	0000000100	00000001000	
1	7	1000000	10000001	100000011	1000000110	1000000110
2		0100000	01000000	010000001	01000000101	010000001010
3		0010000	00100000	001000000	00100000011	001000000101
4		0001000	00010000	000100000	00010000111	000100000111
5		0000100	00001000	000010000	00001000000	000010000000
6		0000010	00000100	000001000	00000100000	000001000000
7		0000001	00000010	000000100	00000010000	000000100000
1	6	100000	1000001	10000011	100000110	1000001110
2		010000	0100000	01000001	0100000101	01000001010
3		001000	0010000	00100000	0010000011	00100000101
4		000100	0001000	00010000	00010000111	000100000111
5		000010	0000100	00001000	0000100000	00001000000
6		000001	0000010	00000100	0000010000	00000100000
1	5	10000	100001	1000011	10000110	100001110
2		01000	010000	0100001	010000101	0100001010
3		00100	001000	0010000	001000011	0010000101
4		00010	000100	0001000	0001000111	00010000111
5		00001	000010	0000100	000010000	0000100000
1	4	1000	10001	100011	1000110	10001110
2		0100	01000	010001	01000101	010001010
3		0010	00100	001000	00100011	001000101
4		0001	00010	000100	000100111	0001001011
1	3	100	1001	10011	100110	1001110
2		010	0100	01001	0100101	01001010
3		001	0010	00100	0010011	00100101
1	2	10	101	1011	10110	101110
2		01	010	0101	010101	010110
1	1	1	11	111	1111	11111

 $n = 11, k = 6$

$i \rightarrow$ 1 2 3 4 5 6
 $g_i \rightarrow$ 10000011101 010000100 00100001011 00010000110 00001000001 00000100000

lists the basis vectors of these groups. The basis vectors have been arranged so that the encoding function has a very simple form described below. Let, for any $s \in S$

$$h = \mu_k(s) \quad h \in H \quad (6-18)$$

and let h^j be the j th most significant bit of h . Then the code word, g , is given by

$$g = \beta(s) \equiv \theta_{n,k}(h) = \sum_{j=1}^k \oplus h^j \cdot g_j \quad (6-19)$$

where $\sum \oplus$ denotes the exclusive-or summation, the dot represents the binary product (or 'and' operation), and g_j 's are the basis vectors of Table 6-2. We use (6-2) for decoding and (6-3) for the calculation of the channel distortion.

Tables 6-3 and 6-4 give the channel distortion, $c(n,k)$, and the total distortion, $t(n,k)$, for the Gaussian density for $p = 10^{-2}$ and $p = 10^{-3}$ for various values of (n,k) . Tables 6-5 and 6-6 give the total distortion for the uniform and the Laplacian densities. The channel distortion for these densities can be easily obtained by subtracting the quantizer distortion given in Table 6-1 from the total distortion. Table 6-7 shows the effects of a proper choice of β . The normal mapping here corresponds to the mapping obtained by the procedure of [78] which is optimum for the uniform density.

We note from (6-2) that the decoder is dependent upon the channel bit-reversal probability p . While from (6-18) and (6-19) we note that the encoder is independent of p for a given G . In practice p might vary from time to time and thus cannot be known exactly. So it becomes necessary to know the robustness of the scheme as p deviates from the

TABLE 6-3

CHANNEL DISTORTION FOR (n,k) BLOCK CODING OF A UNIT VARIANCE (AT THE INPUT OF THE QUANTIZER)
GAUSSIAN RANDOM VARIABLE.

CHANNEL ERROR PROBA- BILITY p	n k	1	2	3	4	5	6	7	8	9	10	11
.01	1	.0252	.0129	.00075	.0004	.00003						
	2		.0483	.0280	.0066	.0018	.0010					
	3			.0656	.0372	.0115	.0033	.0025				
	4				.0765	.0423	.0138	.0056	.0032			
	5					.0821	.0450	.0149	.0062	.0036		
	6						.0856	.0463	.0154	.0064	.0037	.0025
	7							.0923	.0477	.0156	.0068	.0037
	8								.1050	.0508	.0169	.0076
.001	1	.00254	.0013	$.8 \times 10^{-5}$	$.4 \times 10^{-6}$	$< 10^{-6}$						
	2		.0050	.0028	.0005	.00002	.00001					
	3			.0069	.0038	.0010	.00004	.00003				
	4				.0083	.0044	.0012	.00006	.00003			
	5					.0093	.0047	.0013	.0001	.00006		
	6						.0101	.0049	.0014	.0001	.00008	.00005
	7							.0112	.0051	.0014	.00012	.00008
	8								.0143	.0056	.0015	.00015

TABLE 6-4

CHANNEL PLUS QUANTIZER DISTORTION FOR (n,k) BLOCK CODING OF A UNIT VARIANCE (AT THE INPUT OF THE QUANTIZER) GAUSSIAN RANDOM VARIABLE. THE ENTRIES CORRESPONDING TO THE OPTIMUM VALUE OF k ARE UNDERLINED.

CHANNEL ERROR PROBA- BILITY p	n k	1	2	3	4	5	6	7	8	9	10	11
.01	1	.3875	.3751	.3630	.3627	.3623						
	2		<u>.1658</u>	.1455	.1241	.1193	.1185					
	3			<u>.1002</u>	<u>.0717</u>	<u>.0461</u>	<u>.0379</u>	.0371				
	4				.0860	.0518	.0233	.0151	.0127			
	5					.0846	.0475	.0174	.0087	.0061		
	6						.0828	.0469	.0160	.0071	.0043	.0032
	7							.0925	.0478	.0158	.0070	.0039
	8								.1051	.0509	.0168	.0076
.001	1	<u>.3648</u>	.3635	.3623	.3623	.3623						
	2		<u>.1224</u>	.1203	.1180	.1175	.1175					
	3			<u>.0414</u>	.0383	.0355	.0346	.0346				
	4				<u>.0178</u>	.0138	.0107	.0096	.0095			
	5					<u>.0118</u>	<u>.0072</u>	.0038	.0026	.0026		
	6						.0107	.0056	<u>.0020</u>	.0008	.0007	.0007
	7							.0114	.0053	.0016	.0003	.00025
	8								.0143	.0056	.0015	.00021

TABLE 6-5

CHANNEL PLUS QUANTIZER DISTORTION FOR (n, k) BLOCK CODING OF A UNIT VARIANCE (AT THE INPUT OF THE QUANTIZER) UNIFORM RANDOM VARIABLE. THE ENTRIES CORRESPONDING TO THE OPTIMUM VALUE OF k ARE UNDERLINED.

CHANNEL ERROR PROBA- BILITY p	$n \backslash k$	1	2	3	4	5	6	7	8	9	10	11
.01	1	<u>.2797</u>	.2651	.2509	.2505	.2500						
	2		<u>.0996</u>	.0851	.0678	.0640	.0635					
	3			<u>.0546</u>	<u>.0401</u>	<u>.0228</u>	.0182	.0177				
	4				.0434	.0288	.0116	<u>.0073</u>	.0063			
	5					.0406	.0260	.0087	.0045	.0035		
	6						.0398	.0253	.0080	.0038	<u>.0028</u>	<u>.0022</u>
	7							.0397	.0251	.0079	.0036	.0026
	8								.0396	.0251	<u>.00782</u>	.0036
.001	1	.2530	.2515	.2500	.2500	.2500						
	2		.0663	.0648	.0629	.0625	.0625					
	3			<u>.0196</u>	.0181	.0162	.0156	.0156				
	4				<u>.0079</u>	.0064	.0045	.0039	.0039			
	5					<u>.0050</u>	.0035	.0016	.0010	.0010		
	6						<u>.0042</u>	.0027	.0009	.0003	.00028	.00027
	7							<u>.0041</u>	.0026	.0007	.00011	.00010
	8								.0040	.0025	.00066	.000067

TABLE 6-6

CHANNEL PLUS QUANTIZER DISTORTION FOR (n,k) BLOCK CODING OF A UNIT VARIANCE (AT THE INPUT OF THE QUANTIZER) LAPLACIAN RANDOM VARIABLE. THE ENTRIES CORRESPONDING TO THE OPTIMUM VALUE OF k ARE UNDERLINED.

CHANNEL ERROR PROBA- BILITY p	$n \backslash k$	1	2	3	4	5	6	7	8	9	10	11
.01	1	.5194	.5099	.5007	.5004	.5001						
	2		.2389	.2147	.1910	.1857	.1845					
	3			.1505	.1051	.0722	.0613	.0600				
	4				.1390	.0761	.0355	.0243	.0200			
	5					.1423	.0723	.0261	.0133	.0090		
	6						.1415	.0730	.0242	.0109	.0061	.0045
	7							.1409	.0736	.0239	.0102	.0054
	8								.1407	.0739	.0239	.0100
.001	1	.5020	.5010	.5001	.5001	.5001						
	2		.1893	.1867	.1841	.1835	.1835					
	3			.0673	.0620	.0584	.0571	.0571				
	4				.0308	.0223	.0177	.0160	.0160			
	5					.0283	.0117	.0062	.0044	.0043		
	6						.0221	.0094	.0032	.0013	.0012	.00115
	7							.0220	.0091	.0025	.0005	.0004
	8								.0219	.0091	.0024	.0003

TABLE 6-7

EFFECT OF ENCODER MAPPING β . THE REVERSE ENCODING IS OBTAINED BY ASSIGNING THE NORMAL CODE FOR j th QUANTIZER LEVEL TO j' , WHERE j' IS OBTAINED BY REVERSE ORDERING THE BINARY REPRESENTATION OF j . THE ENTRIES SHOW CHANNEL DISTORTION, $c(n,k)$, FOR LAPLACIAN DENSITY, $n = 11$.

MAPPING β	$k \rightarrow$ $p \rightarrow$	6			7			8		
		.1	.01	.001	.1	.01	.001	.1	.01	.001
NORMAL	(1)	.2345	.0034	.00007	.2560	.0051	.00012	.3480	.0091	.00022
REVERSE	(2)	.5651	.1145	.0164	.5996	.1387	.0215	.6028	.1401	.0218
RATIO (2)/(1)		2.410	33.68	234.3	2.330	27.20	179.2	1.73	15.40	99.09

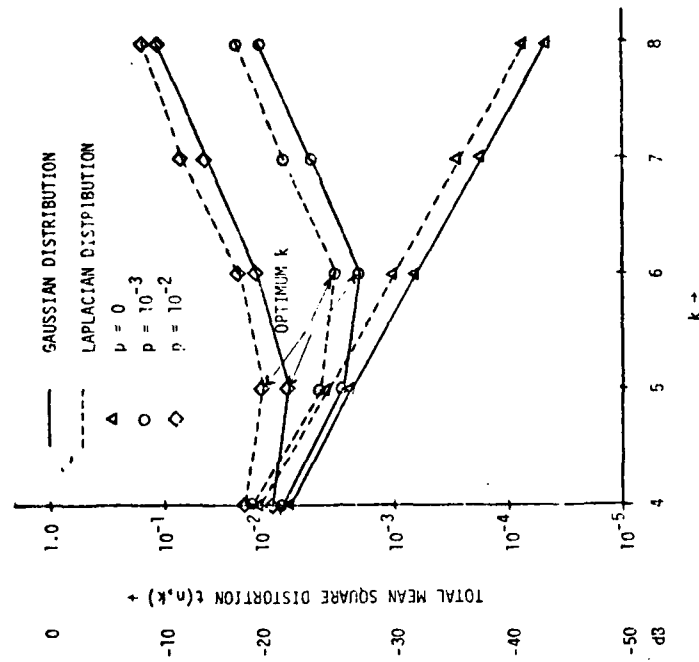
REMARK: The normal mapping gives the maximum protection to the most significant bit of the quantizer output and least protection to the least significant bit, while the reverse encoding does just the opposite. The conventional error correcting codes provide equal protection to all the bits and would perform somewhere in between, but still much inferior to the optimum m.s.e. coding.

design value. Table 6-8 shows that the scheme is indeed quite robust to a wide variation in p .

Figure 6-3 shows the effect of varying k on the total distortion for a fixed rate (or n). The minima of the curves correspond to the optimum value of k . Figure 6-4 gives the distortion-rate functions for various densities with channel optimization. We notice that as the channel becomes noisier, the distortion-rate curves start flattening.

We have also evaluated the performance of the scheme for two important classes of discrete random processes. The first one is a one-dimensional first order Markov process with one step correlation parameter $\rho = .95$. For this process the discrete cosine transform has been known to perform very close to the K-L transform [2]. Hence the matrix Ψ has been chosen to be the discrete Cosine transform [2]. Figure 6-5 shows the distortion-rate curves for this process for Gaussian distribution.

The second class is the 2-D random field with the isotropic covariance model given by (A-2) with $\rho_i = \rho_j = .95$. Once again, we use the discrete Cosine transform, because as shown in Appendix B it performs very close to the K-L transform. Figure 6-6 shows the distortion-rate curves for this process with Gaussian distribution for array sizes of 16×16 and 64×64 . Figure 6-7 shows the bit assignment pattern for the 16×16 array size at 1 bit/sample rate. Figure 6-8 shows the percentage of bits assigned for channel error protection (or redundancy) as a function of rate and array size. We notice that for low channel noise ($p = 10^{-3}$) this percentage is almost constant for different rates as well as array sizes. Even for high levels of channel noise ($p = 10^{-2}$) the variation is not too large. Another noteworthy fact is that the channel



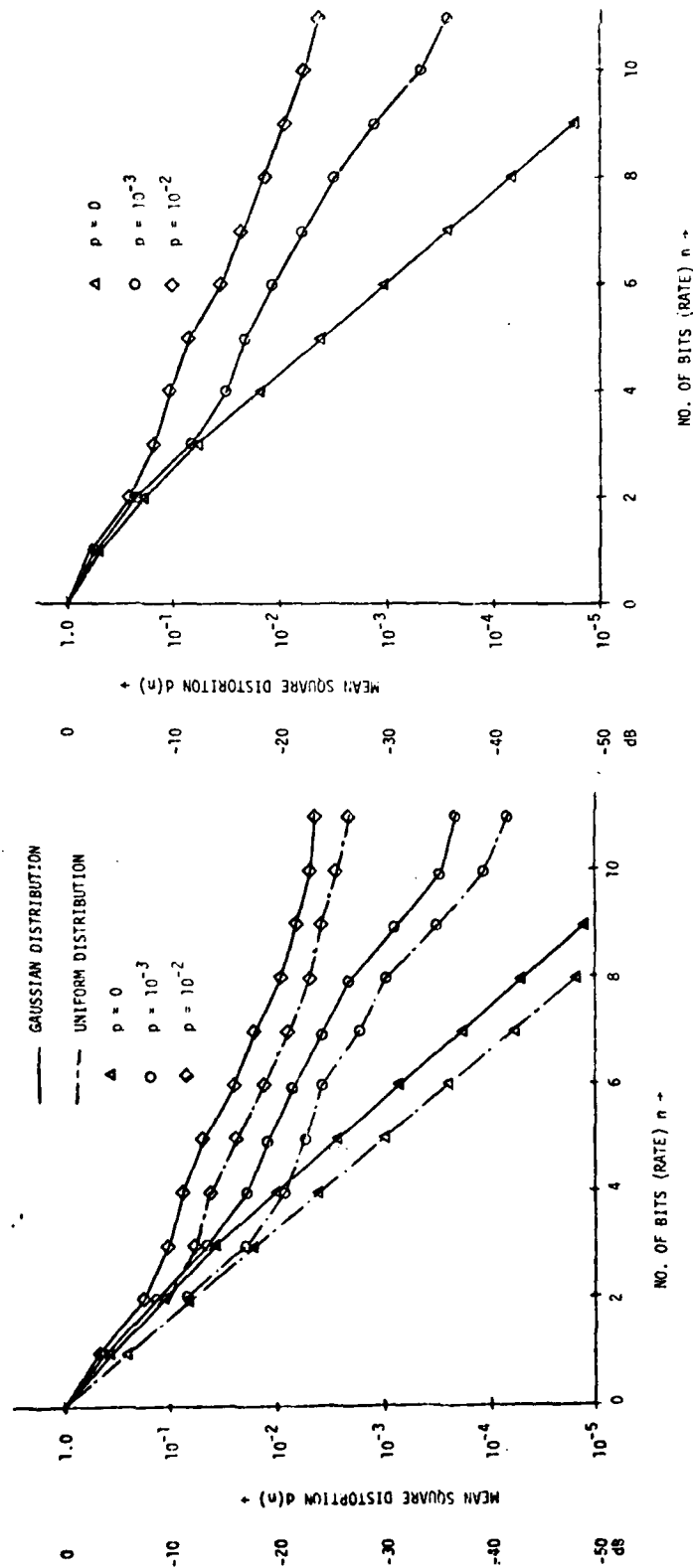
Total Distortion as a Function of Quantization Bits. $n = 8$.

Figure 6-3

TABLE 6-8

CHANNEL DISTORTION FOR (8,6) ENCODER. THE ENTRIES SHOW THAT THE SCHEME IS QUITE ROBUST TO THE FLUCTUATIONS IN CHANNEL STATISTICS.

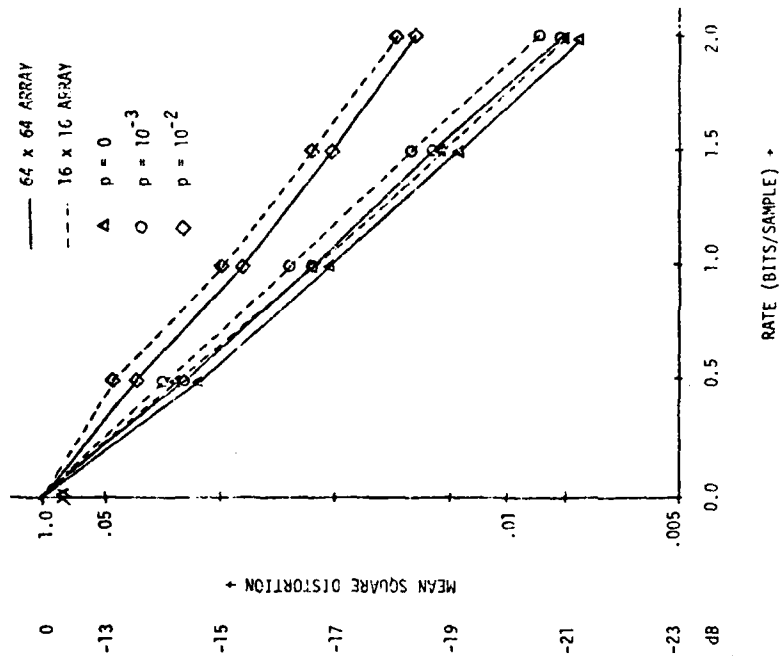
ACTUAL p	OPTIMUM DECODER	DECODER DESIGNED FOR $p = .001$
.0001	.000135	.000140
.0002	.000271	.000274
.0004	.000543	.000545
.0006	.000817	.000818
.0008	.001093	.001094
.001	.001371	.001371
.002	.002783	.002789
.004	.005720	.005783
.006	.008800	.008985
.008	.012014	.012393
.010	.015356	.016004



(a) Gaussian and Uniform Densities

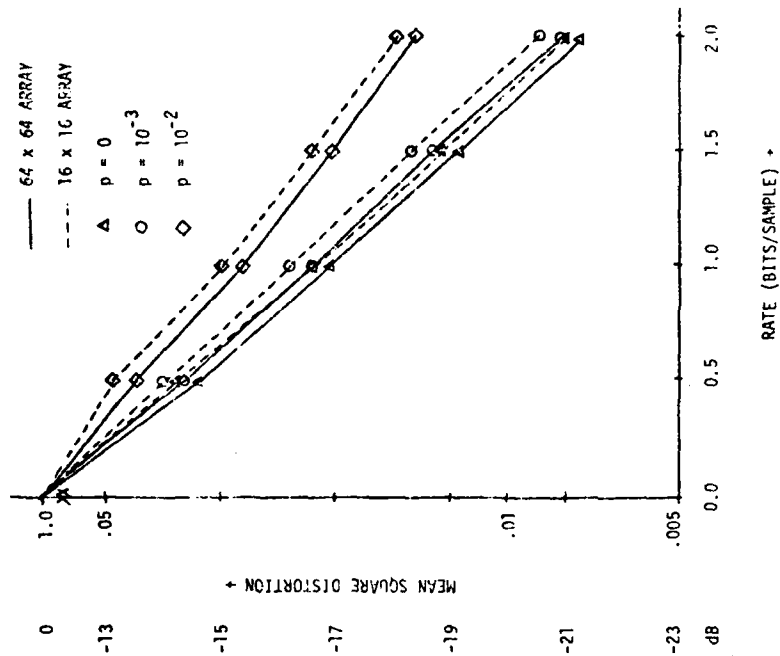
(b) Laplacian Density

Figure 6-4: Distortion-Rate Curves for PCM Transmission of a Random Variable over a Binary Symmetric Channel with Channel Optimization.



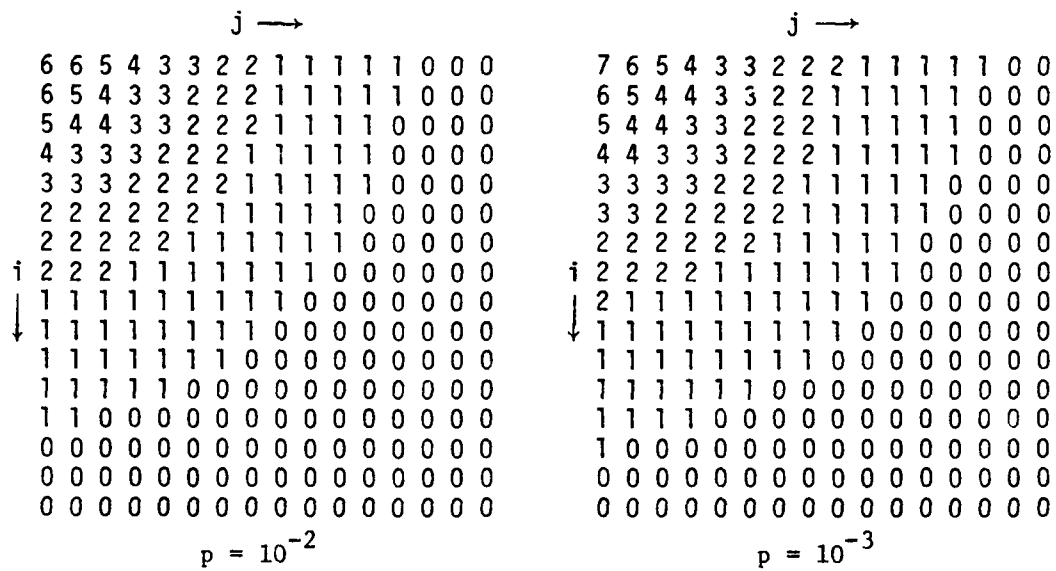
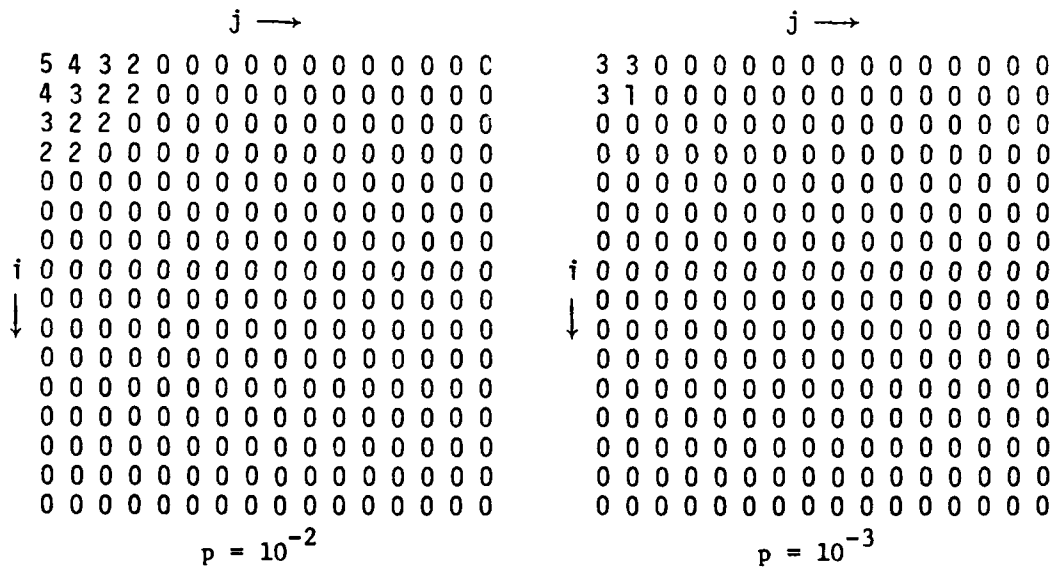
Distortion-Rate Curves for a One-Dimensional First Order Markov Process with Gaussian Distribution Using Channel Optimization. $\rho = .95$.

Figure 6-5



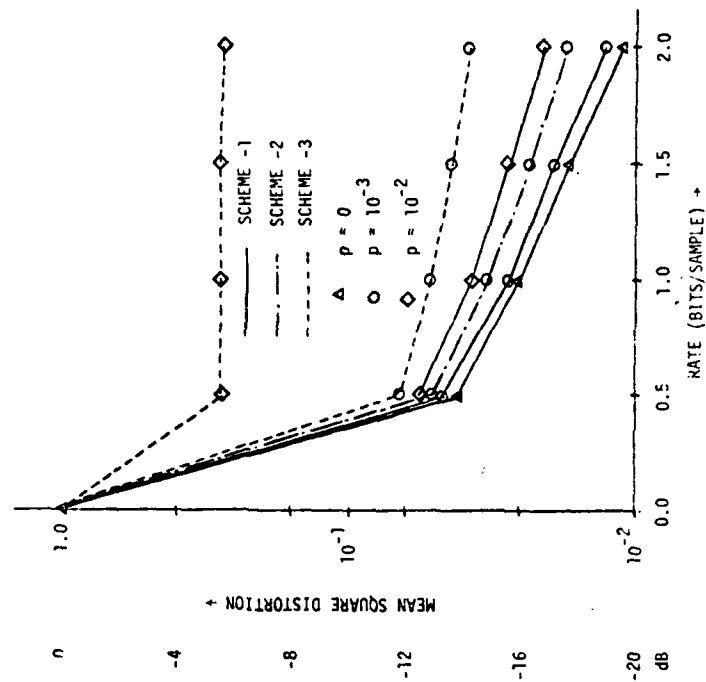
Distortion-Rate Curves for a 2-D Random Process with Gaussian Distribution using Channel Optimization. $\rho_i = \rho_j = .95$.

Figure 6-6

(a) Bit-Assignment for Quantization, $k(i,j)$.(b) Additional Bits for Channel Error Protection, $n(i,j) - k(i,j)$.

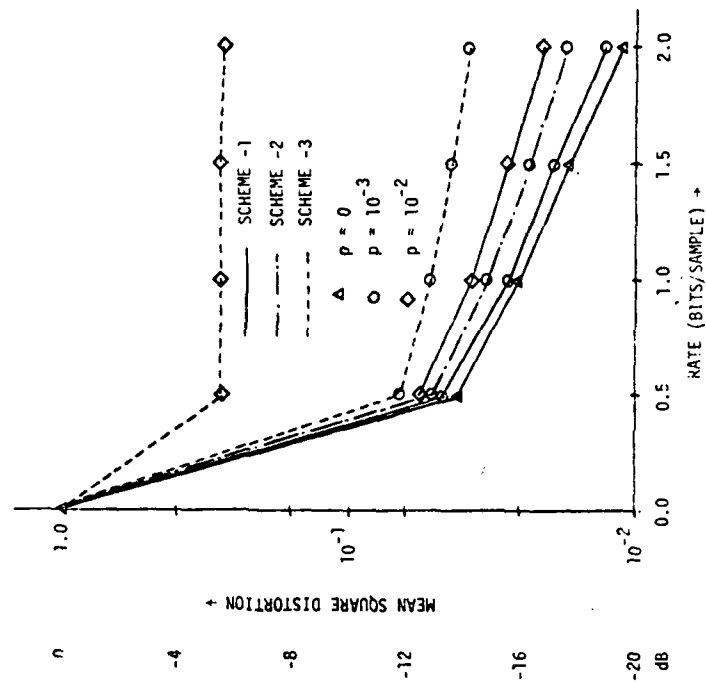
Bit-Assignment for 16×16 Cosine Transform Coding for a 2-D Isotropic Random Field with $\rho_i = \rho_j = .95$. Bit-Rate = 1 Bit/Sample.

Figure 6-7



Percentage of Redundant Bits Assigned for Channel Error Correction for Coding of a 2-D Isotropic Random Process with Gaussian Distribution using Channel Optimization. $\rho_i = \rho_j = .95$.

Figure 6-8



Distortion-Rate Curves for 16×16 Cosine Transform Coding of a 2-D Isotropic Random Field for Various Schemes. $\rho_i = \rho_j = .95$. The Transform Coefficients are Laplacian Distributed.

Figure 6-9

optimization scheme requires much less redundant bits for channel error protection than the conventional error correcting codes would require.

As mentioned in chapter IV, for most images the transform coefficients could be assumed to be Laplacian distributed. So we have also calculated the distortion-rate functions of the 2-D isotropic covariance model for this distribution and compare the channel optimization scheme (Scheme-1) with a (15,11) single error correction encoding scheme (Scheme-2) and a scheme with no error correction (Scheme-3). Figure 6-9 shows the distortion-rate curves for 16×16 array size for these three schemes. The distortion-rate curve for Scheme-2 for $p = 10^{-3}$ has been obtained assuming that the effects of two and more errors in a 15 bit code could be neglected due to their very low probability. Thus the curve is somewhat optimistic and clearly the actual performance of the Scheme-1 relative to the Scheme-2 would be even better than what is shown in Figure 6-9. Roughly we can conclude that the performances of the Scheme-1 for $p = 10^{-2}$, that of the Scheme-2 for $p = 10^{-3}$, and that of the Scheme-3 for $p = 10^{-4}$ are close to each other.

The results of the previous sections were applied for coding a 256×256 Girl image originally digitized to 8 bits/sample. An isotropic covariance model with $\rho_i = \rho_j = .95$ and Laplacian distribution for the transform coefficients were assumed. The array (or sub-block) size of 16×16 and the Cosine transform were chosen in the coding algorithm. The performance of the Scheme-1 and the Scheme-3 were evaluated for $p = 10^{-2}$, 10^{-3} at 1 bit/pixel.

TABLE 6-9

PERFORMANCE OF DATA COMPRESSION SCHEMES AT 1 BIT/PIXEL
FOR COSINE TRANSFORM CODING OF 256 x 256 GIRL IMAGE.
BLOCK SIZE = 16 x 16.

SCHEME	SIGNAL TO NOISE RATIO		
	p=0	p=.001	p=.01
SCHEME -1	31.90 dB	31.40 dB	29.85 dB
SCHEME -3	31.90 dB	25.96 dB	20.05 dB

Table 6-9 gives the signal to noise ratio (SNR) and Figure 6-10 shows the original and the coded images. Figure 6-11 shows various absolute error images amplified ten times. Since the effect of a bit reversal is localized within a sub-block of an image, we call it "blocking effect". From Figure 6-10 we see that for Scheme-1 the performances at $p = 0$ and $p = 10^{-3}$ are almost indistinguishable and at $p = 10^{-2}$ the blocking effects of channel noise are somewhat visible. While for Scheme-3 (which provides no channel noise protection) the blocking effects are quite visible even for $p = 10^{-3}$ and very prominent at $p = 10^{-2}$.

The results of Table 6-9 are in excellent agreement with those of Figure 6-9. Since Figure 6-8 gives the distortion normalized by the variance, while the SNR is normalized by the peak-to-peak signal energy, the SNR for an image could be obtained from Figure 6-9 by subtracting the mean square error in decibels from a constant

$$C = 10 \log_{10} \left\{ \frac{(\text{peak-to-peak signal})^2}{\sigma^2} \right\} \text{ dB}$$

where σ^2 is the variance of the image. For the Girl image we get



(a) Original 256 x 256 Image
8 bits/pixel



(b) Coded at 1 bit/pixel
 $p = 0$



(c) Coded at 1 bit/pixel
 $p = 10^{-3}$, Scheme-3



(d) Coded at 1 bit/pixel
 $p = 10^{-3}$, Scheme-1



(e) Coded at 1 bit/pixel
 $p = 10^{-2}$, Scheme-3



(f) Coded at 1 bit/pixel
 $p = 10^{-2}$, Scheme-1

Images Resulting From 16 x 16 Cosine Transform Coding and Transmission
over a Binary Symmetric Channel.

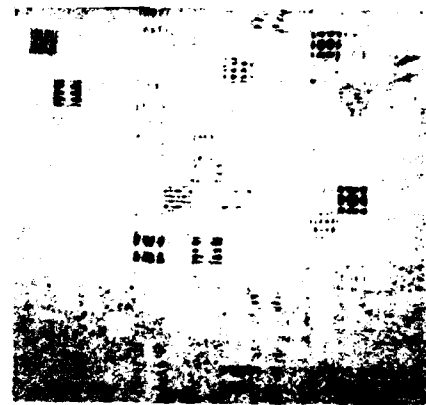
Figure 6-10



(a) Quantization Noise
 $p = 0$



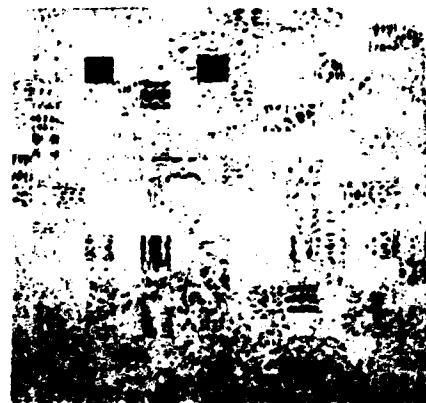
(b) Channel Noise
 $p = 10^{-3}$, Scheme-3



(c) Channel Noise
 $p = 10^{-3}$, Scheme-1



(d) Channel Noise
 $p = 10^{-2}$, Scheme-3



(e) Channel Noise
 $p = 10^{-2}$, Scheme-1

Error Images Corresponding to the Coded Images of the Previous Figure

Figure 6-11

$$C = 10 \log_{10} \left\{ \frac{(255)^2}{1868.76} \right\} \text{ dB}$$

$$= 15.42 \text{ dB.}$$

Thus from Figure 6-9 we obtain the SNR for the Girl image for Scheme-1 at $p = 10^{-3}$ and 1 bit/pixel as

$$\text{SNR} = 15.42 - (-15.48) \text{ dB}$$

$$= 30.90 \text{ dB}$$

which is very close to the actual performance given in Table 6-9 as 31.40 dB. Thus the model used for the data compression of the Girl image seems to be realistic.

It is also apparent by viewing Figure 6-11 that there is a marked difference between the distribution and the visual effects of the two sources of errors, i.e., the quantization and the channel noise. Thus it might be desirable to assign different weights to these errors. This could be easily incorporated in our scheme by defining the total distortion as

$$t(n,k) = q(k) + w_c \cdot c(n,k)$$

and then performing channel optimization as before. A suitable value of weighting coefficient w_c has to be found experimentally.

The concept of channel optimization can also be extended to hybrid coding. This and the application to interframe coding have been left for future research.

CHAPTER VII

SUMMARY, CONCLUSIONS, AND SUGGESTIONS FOR FUTURE WORK

7.1 Summary

The new results presented in this thesis are summarized as follows.

A hypothesis that the temporal dimension of most video motion images consists mainly of a deterministic component, called motion, was presented. A method for the visual characterization of the deterministic component in a stationary mode, based on the temporal cross-sections, was described. A piecewise linear translation model for the motion trajectory estimation was developed. Based on this model, some simple relationships to calculate statistical parameters of the random component were derived.

A new technique for efficient estimation and coding of the deterministic component was presented. The experimental results of application of this technique to actual image data (Head and Shoulders and Chemical Plant) show that it gives very good estimates and that the piecewise linear translation approximation on a sub-block (of suitable size) basis is reasonably good.

The registration of successive frames, called motion compensation, results in a tremendous improvement in prediction (about 10 to 12 dB decrease in interframe variance), and the remaining motion uncertainty in the areas of motion is approximately uniformly distributed between 0 to 0.5 pixels along both the spatial axes. This high degree of registration results in temporal bandwidth reduction, and permits reducing the

sampling rate of the temporal axis (i.e., frame skipping). The missing samples (or frames) can be fairly accurately reproduced by zeroth or first order linear interpolation.

Simple first order Markov covariance models, e.g., separable, result in very poor performance for a transform coder (interframe as well as intraframe) compared with the measured statistical models. Also, the simple (or non-adaptive) transform coders, which are based on approximation of image random process by a wide sense stationary process, result in poor performance for motion images (mostly in reproduction of sharp edges and motion). A significant improvement can be achieved by an adaptive scheme which approximates the nonstationary process by four piecewise stationary processes. However, for the biomedical projection images the assumption of wide sense stationarity is reasonable.

Comparison of the non-adaptive intraframe and interframe schemes for video motion images shows that the mean square performance of the interframe scheme having a sub-block size of $16 \times 16 \times 16$ can be matched by an intraframe scheme having a sub-block size of 64×64 (the total array size for both are the same).

The x-ray projection images have high correlation and the interframe transform coding of these results in very high compression. The effect of the distortion in the projection images on the reconstruction of the 3-D object has been evaluated by reconstructing some transaxial cross-sections (or levels) of the object. The results show that high compression ratios are achievable on these images.

The performance of a hybrid scheme for the video motion images can be significantly improved by adapting the bit-rates and the statistics to the local variations in the spatial and temporal characteristics. A

simple criterion for this adaptation is proposed. Once again, this is achieved by approximating the nonstationary process by four piecewise stationary processes.

The incorporation of motion compensation and 2:1 subsampling of temporal axis results in further significant improvement in the performance of the adaptive hybrid coding. For the Head and Shoulders images high quality images (SNR = 37 dB) are obtained at .125 bit/pixel or a compression ratio of 64 is realized.

The adaptive hybrid coding (without motion compensation) results in very high compression ratios for the angiocardigram x-ray motion images. Compression ratios of 32-128 seem to be realizable based on the evaluation of still images. Spatial statistics of these images are represented very well by stationary models.

A method for the joint optimization of source coding and channel coding for PCM transmission over noisy channels has been presented. It was shown how this method could be applied to image transform coding. The rate distortion curves and the experimental results on images show that this method performs significantly better than the conventional error correcting codes or schemes with no channel protection. For example, at 1 bit rate and channel error probability 10^{-2} , the proposed algorithm improves the performance of an ordinary transform coder (designed for noise free channel) by almost 10 dB.

The performance of the K-L, Cosine, Sine, Fourier, and Hadamard transforms for several commonly used intraframe nonseparable covariance models have been compared for an array size of 8×8 . The results indicate that for all these models (isotropic [53], NC1 [34], and measured covariance of a Girl image) the Cosine transform performs very close to

the optimum K-L transform, and the remaining transforms perform close to each other. The performance of the K-L is about .05 dB better than the Cosine and about 1 dB better than the remaining transforms. Earlier results have shown the near optimality of the Cosine transform for separable covariance model only.

7.2 Conclusions and Recommendations for Future Work

Based on the results and experimental evidence presented in this thesis, we make the following conclusions and recommendations for further investigation.

For multiframe motion images considered here, the motion between successive frames can be very closely approximated by piecewise linear translation of sub-blocks of size 8×8 to 32×32 with an average accuracy of .25 pixel.

The interpolation of skipped frames along motion trajectory (obtained by above approximation) results in excellent encoding of the skipped frames. Thus, we conclude that the bandwidth of the temporal domain can be significantly reduced by motion compensation.

The logarithmic search method of direction of minimum distortion (DMD) could also be useful in many other applications of image registration, e.g., terminal guidance, template matching. This will be a subject of our future research.

The performance of transform coding is highly dependent on the statistical model, especially at high bit-rates. Measuring the statistics in transform domain results in a significant improvement in performance (2-4 dB). We have not addressed the question of how often the

statistics need to be measured for a given application. It has been left for future investigation.

The mean square performance of the nonadaptive intraframe and interframe transform coding schemes is comparable for equal total array size of the sub-block, thus making the interframe transform coding unattractive for motion images. Further investigation is required to establish the effect of array size on coder performance.

The adaptive variable bit-rate transform and hybrid coders have much better performance. The result is improved by 4 dB over the nonadaptive schemes, and the motion and the sharp features are better reproduced.

Motion compensation and alternate frame skipping, with interpolation of skipped frames along motion trajectory, results in a further compression gain by a factor of two. Higher compression gain seems likely by further reducing the sampling along of the temporal axis and interpolation along motion trajectory. Although we have only applied the motion compensation method of chapter II to hybrid coding, this and subsampling of temporal axis can also be used with the predictive coding schemes of chapter III and similar gains are expected. Application of motion compensation to 3-D transform coding seems to be difficult.

The joint optimization of the source coding and the channel coding results in significant improvement in performance of a coding scheme. The concept of channel optimization for PCM transmission can be easily extended to DPCM and thus to the hybrid coding methods.

The cosine transform performs very close to the optimal K-L and its many computational advantages over K-L makes it a better choice for image data compression for a variety of random fields.

Some of the techniques of video bandwidth compression can be applied to the biomedical x-ray images with very high compression. Further research in this area is needed to more qualitatively evaluate the effects of distortion on the medically useful information.

APPENDIX A

MODELING OF INTRAFRAME IMAGE STATISTICS

A.1 Covariance Models for 2-D Images:

The second order statistics of images are required for many image processing applications, e.g., restoration, and coding. Assuming that the images belong to 2-D stationary random fields, a widely used model for image covariances is the separable model given by

$$r_{m,n} \triangleq E[u_{k,\ell} u_{k+m,\ell+n}] = (\rho_i)^m (\rho_j)^n \quad (A-1)$$

$$m, n \geq 0, |\rho_i| < 1, |\rho_j| < 1$$

where $u_{i,j}$ is the intensity of (i,j) th pixel and ρ_i and ρ_j are one step correlation parameters along indices i and j . Without loss of generality we have assumed images having zero mean and unity variance.

Although the model of (A-1) results in a very simple mathematical analysis, it is known to be a poor approximation of the actual image covariances [34]. Another image model, which is called isotropic covariance model [53,66], is known to be a better approximation for most images but has not been used widely so far because of resulting difficulties in analysis. It is given by

$$r_{m,n} = \exp\left\{-\sqrt{\alpha_i m^2 + \alpha_j n^2}\right\} \quad (A-2)$$

$$m, n \geq 0$$

where

$$\alpha_i = (\ln\{\rho_i\})^2; \alpha_j = (\ln\{\rho_j\})^2$$

and $\rho_i = \rho_j$ if the images are sampled at the same rate along both the axes.

Equation (A-2) then simply means that the correlation between any two image points is an exponentially decreasing function of their geometric distance, while in (A-1) it is the sum of the horizontal and vertical distances. From this statement it is clear that (A-2) would be a better model for most images. In [35] it has been demonstrated that the models which closely approximate (A-2) give much better performance in filtering images than that of (A-1).

A model based on a finite difference approximation of an elliptical partial differential equation, reported in [34], and referred as NC1 model [34,35], has been found very useful in modeling image statistics. It is a four point nearest neighbor non-causal (NC) model represented by the relationship

$$u_{i,j} = (u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1}) + \epsilon_{i,j} \quad (A-3)$$

where $\{\epsilon_{i,j}\}$ is a zero mean, moving average field whose covariance function is

$$E[\epsilon_{i,j} \epsilon_{m,n}] = \beta^2 (\delta_{i-m} \delta_{j-n} - \alpha \gamma \delta_{i-m+1} \delta_{j-n+1}) \quad (A-4)$$

and δ is the Kronecker delta function. Suitable values of α , β^2 and γ could be found for a class of images. The application of this model for filtering and data compression could be found in [35] and [75] respectively. The calculation of covariances generated by this model is described in [34].

Sometimes a direct model of covariances is obtained by measuring these quantities for a given image data as follows. Let $K \times L$ be the size of a window over which the covariances are desired and $M \times N$ be the size of data array, U , such that

$$K \ll M \quad \text{and} \quad L \ll N .$$

Then, we define

$$r_{m,n} = \frac{\sum_{i=1}^{M-K} \sum_{j=1}^{N-L} u_{i,j} u_{i+m,j+n}}{\sum_{i=1}^{M-K} \sum_{j=1}^{N-L} u_{i,j}^2} \quad (\text{A-5})$$

$$0 \leq m \leq K-1, \quad 0 \leq n \leq L-1 .$$

A.2 Computation of Transform Coefficient Variances:

For intraframe transform coding, as well as for interframe hybrid coding, we need to know the statistics of images in the transform domain, particularly the variances of the transform coefficients. Let U denote an $M \times N$ block of an image, ψ_L denote an $L \times L$ unitary transform, V denote the $M \times N$ array of the transform coefficients of U , and W the array of the variances of transform coefficients. Then

$$V = \psi_M U \psi_N^T$$

and
$$W = E[(v_{i,j})^2] \quad 1 \leq i \leq M, \quad 1 \leq j \leq N .$$

Let bar on the top of an array represent lexicographic ordering of the elements into a one dimensional array and $R = \{r_{i,j} : 0 \leq i \leq M-1, 0 \leq j \leq N-1\}$ be an $M \times N$ covariance matrix of the image random field. We wish to find the elements of W , given R . From the above definitions, it could be easily seen that

$$\bar{V} = (\psi_M \otimes \psi_N) \bar{U} \quad (\text{A-6})$$

and
$$\bar{W} = \text{Diag.}\{E[\bar{V} \bar{V}^T]\} \quad (\text{A-7})$$

where \otimes denotes the Kronecker product of matrices and $\text{Diag}\{B\}$ represents a one dimensional array containing the diagonal elements of a square matrix

B. From (A-6) and (A-7) we have

$$\bar{W} = \text{Diag}\{(\psi_M \otimes \psi_N) E [\bar{U} \bar{U}^T] (\psi_N^T \otimes \psi_M^T)\}$$

or
$$\bar{W} = \text{Diag}\{(\psi_M \otimes \psi_M) \mathcal{R} (\psi_N^T \otimes \psi_N^T)\} \quad (\text{A-8})$$

where \mathcal{R} is a $N \times N$ symmetric block Toeplitz matrix whose each element is an $M \times M$ symmetric Toeplitz matrix. The elements of \mathcal{R} are given by

$$\mathcal{R}_{i,j;k,l} = r_{|k-l|, |i-j|} \quad 1 \leq i, j \leq N, \quad 1 \leq k, l \leq M \quad (\text{A-9})$$

where the first two subscripts of \mathcal{R} refer to the addresses of the blocks and the last two refer to the addresses of the elements within a block.

Thus, one can calculate the transform coefficient variances by appropriately taking the transform of \mathcal{R} . We have also found an efficient algorithm for computing (A-8) which exploits the Toeplitz structure of \mathcal{R} and the fact that only diagonal elements of its transform are needed. This will be published elsewhere.

Table A-1 shows the cosine transform coefficient variances for a sub-block size of 16×16 measured over the 16 frames of Head and Shoulders data set. Each sub-block was first transformed by a discrete Cosine transform and then for each transform coefficient the variance was measured over all the data sub-blocks. Table A-2 shows a 2-D 16×16 covariance matrix, R , corresponding to the model of (A-2) and Table A-3 shows its corresponding Cosine transform coefficient variance matrix, W . Comparing the corresponding entries of Tables A-1 and A-3, the ratio is not too far from

unity for the lower order coefficients, while for higher order coefficients the ratio is too far deviated from unity.

The consequence of the above is that at lower bit-rates, where no bits are assigned to most higher order coefficients, the performance of the coders using variances of Tables A-1 or A-3 would be very close and hence (A-2) is a good model. But at higher bit-rates, the variance distribution of Table A-3 would tend to assign bits to higher order coefficients unnecessarily. We have found experimentally that for high resolution smooth images, which have very low variance for higher order coefficients, a correction factor applied to the transform coefficient variances, resulting from the model of (A-2), improves the performance at high bit-rates considerably. One such correction factor, for Head and Shoulders, data, is given by

$$w'_{i,j} = hw_{i,j}(\rho) \quad b \left\{ \left(\frac{i-1}{M} \right)^2 + \left(\frac{j-1}{N} \right)^2 \right\}, \quad b = 50 \quad (A-10)$$

where $\rho = (\rho_i + \rho_j)/2$

and h is chosen such that

$$\sum_{i=1}^M \sum_{j=1}^N w'_{i,j} = \sum_{i=1}^M \sum_{j=1}^N w_{i,j}.$$

Table A-4 shows the matrix of Table A-3 with the above correction factor. We can see that the entries of Tables A-1 and A-4 are close and hence the model of (A-2) with the correction factor of (A-10) is a better model for coding the images belonging to the same class as the Head and Shoulders images than (A-2) with no correction. However, this correction factor is data dependent, but a suitable value of parameter b in (A-10) could be found for other data.

TABLE A-1

TRANSFORM COEFFICIENT VARIANCES FOR 16×16 SUB-BLOCK SIZE COSINE TRANSFORM MEASURED OVER THE 16 FRAMES OF HEAD AND SHOULDERS DATA.

183.240	24.240	6.423	2.140	.839	.583	.244	.149	.111	.084	.063	.043	.028	.016	.007	.002
20.928	3.942	1.509	.744	.431	.144	.064	.028	.016	.009	.005	.003	.001	.001	.000	.000
3.630	1.146	.650	.416	.241	.111	.053	.025	.011	.007	.004	.002	.001	.001	.000	.000
1.265	.352	.256	.151	.099	.073	.040	.021	.011	.006	.003	.002	.001	.000	.000	.000
.365	.142	.106	.084	.045	.032	.019	.015	.008	.005	.003	.001	.001	.000	.000	.000
.145	.058	.044	.032	.027	.017	.011	.006	.005	.003	.002	.001	.001	.000	.000	.000
.057	.024	.023	.016	.013	.009	.006	.004	.003	.002	.001	.001	.000	.000	.000	.000
.025	.013	.011	.008	.007	.004	.004	.002	.001	.001	.001	.000	.000	.000	.000	.000
.012	.007	.006	.004	.004	.003	.002	.002	.001	.001	.001	.000	.000	.000	.000	.000
.006	.003	.003	.003	.002	.001	.001	.001	.001	.000	.000	.000	.000	.000	.000	.000
.004	.002	.002	.002	.002	.001	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000
.002	.001	.001	.002	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
.001	.001	.001	.001	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
.001	.000	.001	.001	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.001	.001	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.001	.001	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000

TABLE A-2

16×16 COVARIANCE MATRIX, R, FOR A TWO-DIMENSIONAL ISOTROPIC STATIONARY RANDOM FIELD FOR $\rho_i = .955$, $\rho_j = .945$.

1.000	.945	.893	.844	.797	.754	.712	.673	.636	.601	.568	.537	.507	.479	.453	.428
.955	.930	.885	.839	.794	.751	.710	.671	.635	.600	.567	.536	.506	.479	.452	.427
.912	.898	.864	.824	.783	.743	.703	.666	.630	.596	.564	.533	.504	.477	.451	.426
.871	.861	.836	.803	.767	.730	.693	.657	.623	.590	.559	.529	.500	.473	.444	.423
.832	.825	.806	.778	.747	.714	.680	.646	.613	.582	.552	.523	.495	.469	.443	.420
.794	.789	.774	.751	.724	.694	.664	.633	.602	.572	.543	.515	.488	.463	.438	.415
.759	.754	.742	.723	.700	.673	.646	.617	.588	.560	.533	.506	.481	.456	.432	.410
.724	.721	.711	.695	.674	.651	.626	.600	.574	.547	.521	.496	.472	.448	.425	.403
.692	.689	.680	.667	.649	.628	.606	.582	.558	.533	.509	.485	.462	.436	.418	.397
.661	.658	.651	.639	.624	.605	.585	.564	.541	.519	.496	.473	.451	.430	.409	.389
.631	.629	.622	.612	.599	.583	.564	.545	.524	.503	.482	.461	.440	.420	.400	.381
.603	.601	.595	.586	.574	.560	.544	.526	.507	.488	.468	.448	.429	.409	.389	.372
.575	.574	.569	.561	.550	.538	.523	.507	.490	.472	.453	.435	.417	.399	.381	.363
.550	.548	.544	.537	.527	.516	.503	.488	.472	.456	.439	.422	.405	.387	.371	.354
.525	.524	.520	.513	.505	.495	.483	.469	.455	.440	.424	.408	.392	.376	.360	.345
.501	.500	.497	.491	.483	.474	.463	.451	.438	.424	.410	.395	.380	.365	.350	.335

TABLE A-3

16 × 16 COSINE TRANSFORM COEFFICIENT VARIANCES CORRESPONDING TO THE COVARIANCE MATRIX OF TABLE A-2.

170.335	23.137	5.748	1.883	.874	.456	.277	.179	.126	.092	.072	.058	.049	.043	.039	.037
17.237	5.935	2.602	1.175	.620	.359	.229	.155	.112	.084	.067	.055	.047	.042	.038	.036
3.667	2.130	1.299	.749	.452	.287	.194	.137	.102	.079	.063	.053	.045	.041	.038	.036
1.133	.832	.638	.441	.305	.215	.155	.115	.089	.071	.058	.049	.043	.039	.036	.035
.515	.412	.341	.267	.205	.157	.121	.095	.076	.063	.053	.046	.041	.037	.035	.033
.246	.229	.202	.171	.141	.115	.094	.077	.065	.055	.047	.042	.038	.035	.033	.032
.162	.144	.131	.116	.101	.086	.074	.063	.055	.048	.042	.038	.035	.033	.031	.030
.105	.097	.091	.083	.075	.067	.059	.052	.047	.042	.038	.035	.032	.031	.029	.029
.075	.070	.067	.063	.058	.053	.048	.044	.040	.037	.034	.032	.029	.029	.028	.027
.056	.054	.052	.049	.047	.044	.041	.038	.035	.033	.031	.029	.028	.027	.026	.026
.045	.043	.042	.041	.039	.037	.035	.033	.031	.030	.028	.027	.026	.025	.025	.025
.037	.036	.035	.035	.033	.032	.031	.030	.028	.027	.026	.025	.025	.024	.024	.023
.032	.031	.031	.030	.030	.029	.028	.027	.026	.025	.025	.024	.024	.023	.023	.023
.029	.028	.028	.028	.027	.027	.026	.025	.025	.024	.024	.023	.023	.022	.022	.022
.027	.026	.026	.026	.026	.025	.025	.024	.024	.023	.023	.022	.022	.022	.022	.021
.025	.025	.025	.025	.025	.024	.024	.023	.023	.023	.022	.022	.022	.021	.021	.021

TABLE A-4

TRANSFORM COEFFICIENT VARIANCES OF THE ABOVE TABLE AFTER CORRECTION FACTOR OF EQN. (A-10).

177.361	23.845	5.745	1.788	.772	.367	.200	.113	.068	.042	.027	.017	.012	.008	.005	.004
17.765	6.055	2.574	1.104	.542	.287	.163	.097	.060	.038	.025	.016	.011	.008	.005	.004
3.665	2.107	1.246	.682	.383	.222	.134	.083	.053	.034	.023	.015	.010	.007	.005	.004
1.076	.782	.572	.382	.246	.158	.102	.066	.044	.029	.020	.014	.009	.007	.005	.004
.455	.361	.289	.215	.154	.107	.074	.051	.035	.024	.017	.012	.008	.006	.004	.003
.214	.182	.156	.126	.096	.072	.052	.038	.027	.019	.014	.010	.007	.005	.004	.003
.117	.102	.091	.076	.061	.048	.037	.028	.020	.015	.011	.008	.006	.004	.003	.002
.046	.040	.035	.048	.040	.032	.026	.020	.015	.011	.009	.006	.005	.004	.002	.002
.041	.036	.035	.031	.027	.022	.018	.014	.011	.009	.007	.005	.004	.003	.002	.001
.026	.024	.023	.020	.018	.015	.013	.010	.008	.007	.005	.004	.003	.002	.002	.001
.017	.016	.015	.014	.012	.011	.009	.007	.006	.005	.004	.003	.002	.002	.001	.001
.011	.011	.010	.009	.009	.008	.006	.005	.004	.004	.003	.002	.002	.001	.001	.001
.008	.007	.007	.007	.006	.005	.005	.004	.003	.003	.002	.002	.001	.001	.001	.001
.005	.005	.005	.005	.004	.004	.003	.003	.002	.002	.002	.001	.001	.001	.001	.000
.004	.004	.003	.003	.003	.003	.002	.002	.002	.001	.001	.001	.001	.001	.000	.000
.003	.003	.002	.002	.002	.002	.002	.001	.001	.001	.001	.001	.001	.000	.000	.000

APPENDIX B

COMPARISONS OF 2-D TRANSFORMS

Several discrete unitary transforms have been used for intraframe and interframe transform coding of images. These are Karhunen-Loeve (or K-L), Fourier, Cosine, Sine, Hadamard, Haar, Slant, etc. Of these the K-L transform is the optimum transform for data compression (the performance criterion is discussed in section B.1) and is dependent on the statistics of the data. The remaining transforms are data independent and also have FFT type fast computational algorithms. For these reasons, the others are preferred over the K-L in practice.

For a class of one-dimensional signals, i.e., first order Markov process with high correlation, the discrete Cosine transform (or DCT) is known to perform very close to the K-L transform [2,33]. Since the 2-D DCT is defined as a separable product (i.e., Kronecker product) of the one-dimensional DCT, it follows from the above that it will perform very close to the 2-D K-L transform for a 2-D separable first order Markov field given by (A-1), for highly correlated data such as images. Although the separable model of (A-1) has been used for data compression [67,68,87], for reasons discussed in Appendix A, nonseparable models are preferable in many cases. Therefore, the nonseparable models described by (A-2), (A-3), (A-4), (A-5) and others have been used for data compression [49,53, 67,68,75] with better results than the separable model. The most commonly used transforms in these studies are the Cosine and the Hadamard. The former for its better performance and the latter for its simplicity. However, no theoretical or experimental evidence exists for the relative performance of various transforms for nonseparable fields.

We have done some evaluations of the performance of K-L, Cosine, Sine, Fourier, and Hadamard transforms for a number of commonly used nonseparable covariances. Our results show that the Cosine transform performs extremely close to the optimum K-L transform.

B.1 Transforms and Their Performance Measure:

Let \bar{U} denote an $MN \times 1$ vector array obtained by lexicographic ordering of a real $M \times N$ 2-D array U . We define its transform by

$$\bar{V} = A\bar{U} \quad (B-1)$$

If the transform matrix A could be written as

$$A = A_1 \otimes A_2 \quad (B-2)$$

where A_1 and A_2 are $M \times M$ and $N \times N$ matrices respectively, then it is called a separable transform and (B-1) could be written as

$$V = A_1 U A_2^T \quad (B-2)$$

where V is an $M \times N$ array obtained by inverse lexicographic ordering of \bar{V} . The K-L transform, characterized by maximum mean square energy compaction property, consists of the eigenvectors of the matrix \mathcal{R} defined by (A-9). Note that the K-L transform A_{K-L} corresponding to an arbitrary covariance matrix \mathcal{R} , which is not separable, is not separable. Let

$$\sigma_k^2 = E[\bar{V}_k^2] \quad 1 \leq k \leq MN$$

and the rows of A be arranged such that

$$\sigma_i^2 \geq \sigma_j^2 \quad \forall_{i < j} \quad (B-4)$$

Note that the sequence $\{\sigma_k^2\}$ consists of the elements of matrix W defined in Appendix A.

We restrict our attention to transforms which are unitary (all the transforms discussed above are unitary), i.e.,

$$A^{-1} = A^{*T}.$$

For the K-L transform, the sequence $\{\sigma_k^2\}$ is nothing but the eigenvalues of \mathcal{R} arranged in descending order.

We define the performance of a transform by a sequence of basis restriction errors $\{b_i: 0 \leq i \leq MN\}$ defined by

$$b_i = \frac{\sum_{k=i+1}^{MN} \sigma_k^2}{\sum_{k=1}^{MN} \sigma_k^2} \quad 0 \leq i \leq MN-1 \quad (B-5)$$

$$b_{MN} = 0.$$

Each b_i represents the normalized minimum mean square error if only i of the transform coefficients are retained. For the K-L transform the sequence $\{b_i\}$ is minimum, i.e.,

$$(b_i)_{K-L} \leq (b_i) \quad (B-6)$$

or

$$\sum_{k=1}^i (\sigma_k^2)_{K-L} \geq \sum_{k=1}^i \sigma_k^2. \quad (B-7)$$

Thus, the K-L transform is optimum in the sense that it minimizes the mean square error when some of the transform coefficients are discarded. Since $\{\sigma_k^2\}$ represents the mean square energy of the transform coefficients, the property (B-7) is called maximum mean square energy compaction property of the K-L transform.

Another performance measure used in data compression applications is the distortion-rate function which is defined below. Let each transform coefficient be independently quantized to a finite number of levels and d_i be the mean square error per unit variance due to the quantization of the i th coefficient of array \bar{V} , then

$$D = \sum_{i=1}^{MN} \sigma_i^2 d_i \quad (B-8)$$

gives the total mean square distortion in a transform coding system with a noiseless channel. Let n_i be the number of bits required to code the output of the i th quantizer. Then the rate is given by

$$R_D = \frac{1}{MN} \sum_{i=1}^{MN} n_i \quad \text{bits/sample,} \quad n_i = \text{integer} \geq 0 \quad (B-9)$$

where the sequence $\{n_i\}$ is chosen such that D in (B-8) is minimized for a fixed R_D . The D vs R_D curves obtained from (B-8) and (B-9) are the distortion-rate functions for an integer bit allocation scheme.

B.2 Experimental Results:

We compare the performance of the K-L, Cosine, Sine, Fourier, and Hadamard discrete transforms which are often considered for data compression. The definition and properties of these and some other transforms could be found in [3,31,58]. We have chosen two block sizes which are of interest in data compression, i.e., 8×8 and 16×16 . For some of the comparisons the complexity of computing the eigenvalues of matrix prohibits sizes larger than 8×8 .

All the above transforms other than Fourier are real, and result in MN real nonredundant transform coefficients for an array size of $M \times N$.

Since the data is assumed to be real, of the MN complex Fourier coefficients, only $(MN/2 + 2)$ are real and $(MN/2 - 2)$ imaginary components are nonredundant (due to symmetry). It is therefore sufficient to consider the variances of these components in obtaining the sequence $\{\sigma_i^2\}$.

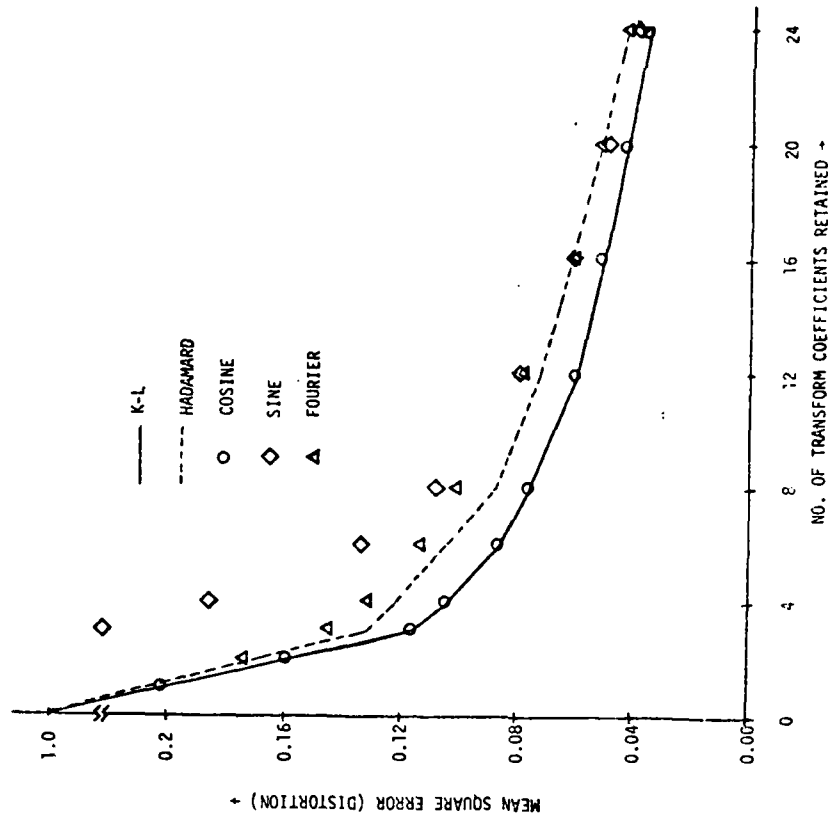
Figures B-1 through B-3 show the basis restriction errors for the above mentioned transforms for $M = 8$, $N = 8$ and the three nonseparable random fields described by (A-2), (A-3), (A-4) and (A-5) in Appendix A. For the isotropic field of (A-2), the values of $\rho_i = \rho_j = .95$ were chosen [53]. For the noncausal NC1 model of (A-3) and (A-4) the values of the parameters were chosen to be

$$\alpha = .2496, \beta^2 = .6744, \gamma = .95.$$

These values were found by a least squares fit of the model and the 16×16 measured covariances for the Girl image shown in Figure 6-10(a) [34,35]. For the measured covariance model of (A-5), the same Girl image data was used.

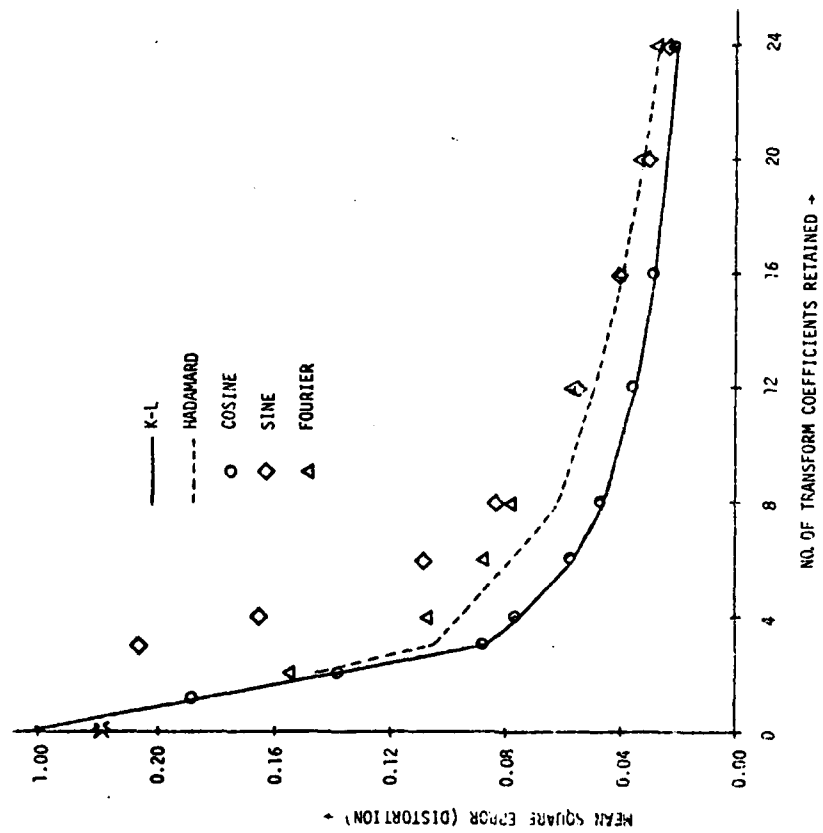
Figure B-4 shows the distortion-rate curves for the isotropic model assuming a Gaussian distribution for the transform coefficients. The distortion has been calculated based on optimum mean square quantization [47] and optimum integer bit allocation (i.e., via integer programming algorithm) of [86]. Figures B-5 and B-6 show the distortion-rate curves for the isotropic model of (A-2) and the separable model of (A-1) respectively, for $\rho_i = \rho_j = .95$, $M = N = 16$. For this array size, the K-L transform was excluded due to computational difficulties. Table B-1 gives the results of Figures B-4 and B-5 in numerical form.

From Figures B-1 thru B-4 we note that the performance of the Cosine transform is very close to the optimum K-L transform for all the



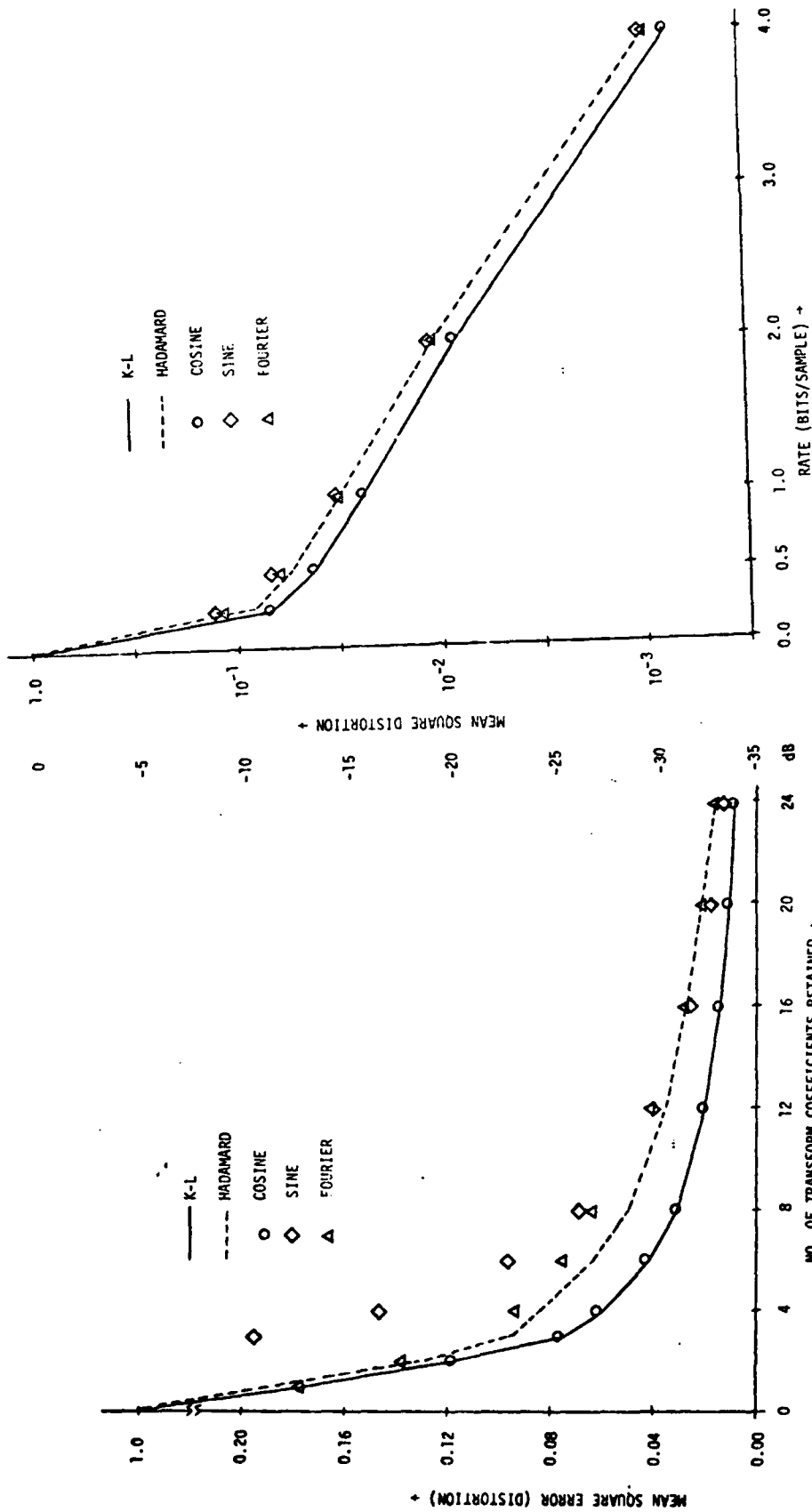
Basis Restriction Error of an 8×8 Transform for the Isotropic Covariance Model. $\rho_1 = \rho_j = .95$.

Figure B-1



Basis Restriction Error of an 8×8 Transform for the NC-1 Model.

Figure B-2



Distortion-Rate Curves for Transform Coding of an 8×8 Array for the Isotropic Covariance Model with Gaussian Distribution. $\rho_1 = \rho_j = .95$.

Figure B-4

Basis Restriction Error of an 8×8 Transform for the Measured Covariance Model for a Girl Image.

Figure B-3

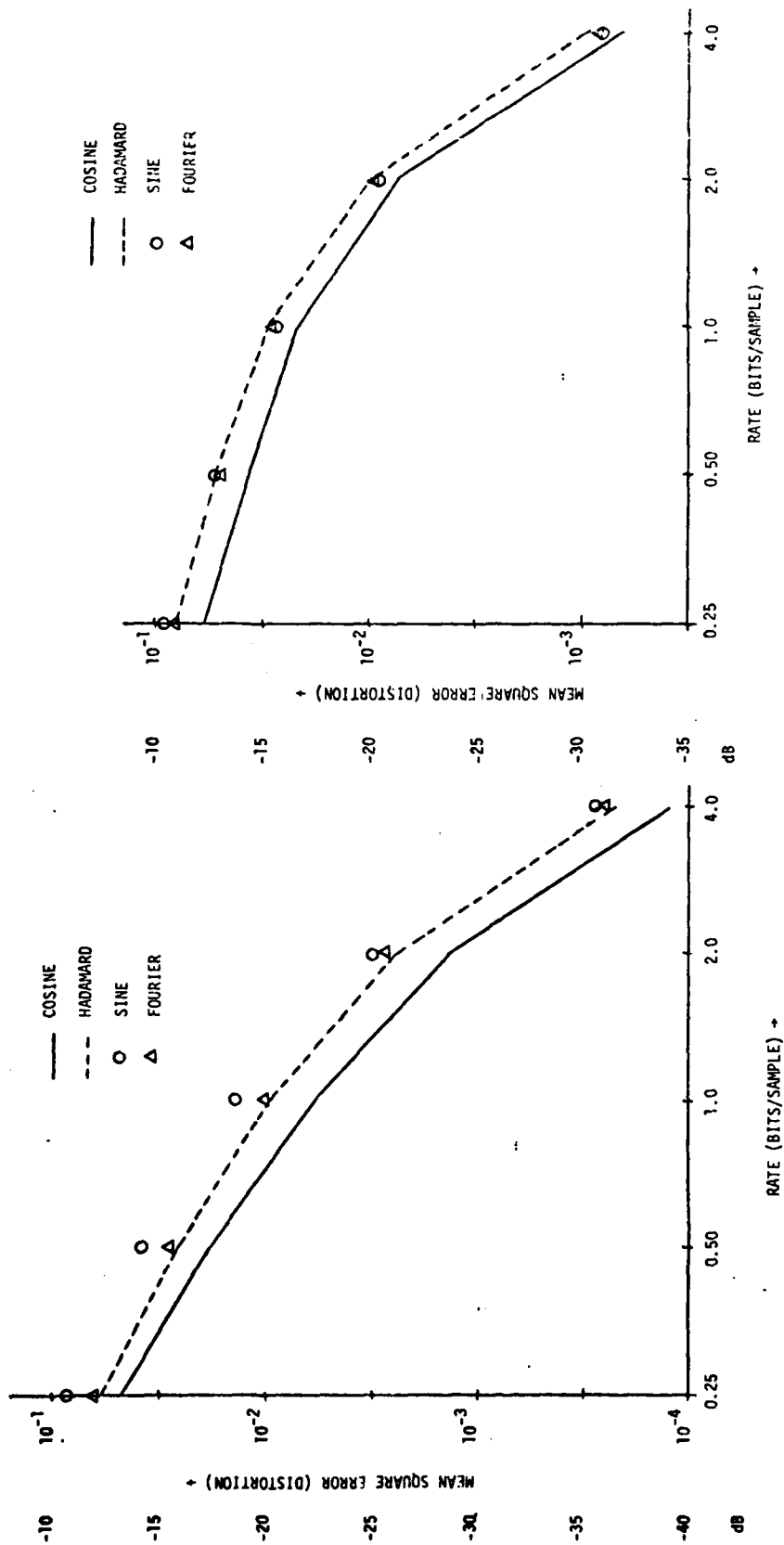


Figure B-5

Distortion-Rate Curves for Transform Coding of a
16 x 16 Array for the Separable Covariance Model with
Gaussian Distribution. $\rho_i = \rho_j = .95$.

Distortion-Rate Curves for Transform Coding of a
16 x 16 Array for the Isotropic Covariance Model
with Gaussian Distribution. $\rho_i = \rho_j = .95$.

Figure B-6

TABLE B-1
 DISTORTION-RATE COMPARISONS OF VARIOUS DISCRETE TRANSFORMS FOR AN ISOTROPIC
 COVARIANCE FIELD WITH $\rho_i = \rho_j = 0.95$, UNIT VARIANCE AND GAUSSIAN DISTRIBUTION.

ARRAY SIZE	RATE BITS/ PIXEL	DISTORTION IN DECIBELS				
		K-L	COSINE	SINE	FOURIER	HADAMARD
8x8	0.25	-11.74	-11.66	- 9.08	-10.15	-10.79
	0.50	-13.82	-13.76	-11.69	-12.27	-12.65
	1.00	-16.24	-16.19	-14.82	-14.99	-15.17
	2.00	-20.95	-20.89	-19.53	-19.73	-19.86
	4.00	-31.61	-31.54	-30.17	-30.44	-30.49
16x16	0.25		-12.35	-10.37	-10.77	-10.99
	0.50		-14.25	-12.82	-12.87	-12.78
	1.00		-16.58	-15.65	-15.52	-15.27
	2.00		-21.26	-20.37	-20.24	-20.01
	4.00		-31.90	-31.00	-30.88	-30.69

three nonseparable models. While the performance of the remaining three is not too close to the K-L but quite close to each other. The performance of the Cosine transform is about .05 dB inferior to the K-L while that of others is about 1.0 dB. At a bit rate of 1 bit/pixel the Cosine would require a rate increase of approximately 1% to match the performance of the K-L while the others would require about 25% increase. From Figures B-5 and B-6 we note that the relative performance does not change much for a slightly larger array size as well as for considerable different models.

Since the Cosine transform can be implemented by a fast algorithm [2,15] and is data independent, its computational advantages over the K-L overwhelm the marginal difference in performance. The performance differences between the Cosine, Sine, Fourier, and the K-L will decrease further as the array size is increased. Since all these sinusoidal transforms are asymptotically equivalent [33].

Thus the prime advantage of Cosine transform coding remains in the common situation where a larger image is coded block by block with typical block size of 16×16 or 8×8 . Finally, we note that the recursive block-coding of random fields via fast K-L transform algorithms [36] achieve rates close to and better (!) than conventional K-L transform block-coding method, by coding the boundary variables of a block separately and exploiting the interblock redundancy represented by the boundary variables. Comparison with these algorithm is not made here and is left as a future study.

REFERENCES AND BIBLIOGRAPHY

1. N. Abramson, Information Theory and Cod'ng. New York: McGraw Hill, 1963.
2. N. Ahmed, et al., "Discrete Cosine Transform," IEEE Trans. on Computers, vol. C-23, pp. 90-93, January 1974.
3. N. Ahmed and K.R. Rao, Orthogonal Transforms for Digital Signal Processing. New York: Springer Verlag, 1975.
4. V.R. Algazi, "Useful Approximations to Optimum Quantization," IEEE Trans. on Commun. Tech., vol. COM-14, pp. 297-301, June 1966.
5. H.C. Andrews and C.L. Patterson, "Singular Value Decomposition Image Coding," IEEE Trans. on Commun., vol. COM-24, pp. 425-432, April 1976.
6. H.H. Bauch, et al., "Picture Coding," (survey of work in West Germany), IEEE Trans. on Commun., vol. COM-22, pp. 1150-1167, September 1974.
7. T. Berger, Rate Distortion Theory: A Mathematical Basis for Data Compression. Englewood Cliffs, N.J.: Prentice-Hall, 1971.
8. E.O. Brigham, The Fast Fourier Transform. Englewood Cliffs, N.J.: Prentice-Hall, 1974.
9. S. Brofferio and F. Rocca, "Interframe Redundancy Reduction of Video Signals Generated by Translating Objects," IEEE Trans. on Commun., vol. COM-25, pp.448-455, April 1977.
10. J. Burgmeier, "Three Dimensional DPCM with Entropy Coding and Adaptive Filtering," Picture Coding Symposium, Asilomar, California, Jan 28-30, 1976.
11. C. Cafforio and F. Rocca, "Method for Measuring Small Displacements of Television Images," IEEE Trans. on Inform. Theory, vol. IT-22, pp. 573-579, September 1976.
12. P. Camana, "Video-Bandwidth Compression: A Study in Tradeoffs," IEEE Spectrum, vol. 16, pp. 24-29, June 1979.
13. J.C. Candy, et al., "Transmitting Television as Clusters of Frame to Frame Differences," B.S.T.J., vol. 50, pp. 1889-1917, July 1971.
14. K.Y. Chang and R.W. Donaldson, "Nonadaptive DPCM Transmission of Monochrome Pictures over Noisy Channels," IEEE Trans. on Commun., vol. COM-24, pp. 173-183, February 1976.
15. W.H. Chen, et al., "A Fast Computational Algorithm for the Discrete Cosine Transform," IEEE Trans. on Commun., vol. COM-25, pp. 1004-1009, September 1977.

16. D.L. Cohn and J.L. Melsa, "The Relationship Between an Adaptive Quantizer and a Variance Estimator," IEEE Trans. on Inform. Theory, vol. IT-21, pp. 669-671, November 1975.
17. D.J. Connor and J.O. Limb, "Properties of Frame Difference Signal Generated by Moving Images," IEEE Trans. on Commun., vol. COM-22 pp. 1564-1575, October 1974.
18. T.R. Crimmins and H.M. Horwitz, "Mean-square Error Optimum Coset Leaders for Group Codes," IEEE Trans. on Inform. Theory, vol. IT-16 pp. 429-432, July 1970.
19. T.R. Crimmins, et al., "Minimization of Mean-square Error for Data Transmission via Group Codes," IEEE Trans. on Inform. Theory, vol. IT-15, pp. 72-78, January 1969.
20. L.D. Davisson, "Rate-Distortion Theory and Application," Proc. IEEE, vol. 60, pp. 800-808, July 1972.
21. J.I. Gimlet, "Use of 'Activity' Classes in Adaptive Transform Image Coding," IEEE Trans. on Commun., vol. COM-23, pp. 785-786, July 1975.
22. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Trans. on Commun., vol. COM-22, pp. 614-624, May 1974.
23. B.G. Haskell, "Entropy Measurement for Nonadaptive and Adaptive Frame-to-Frame, Linear Predictive Coding of Videotelephone Signals," B.S.T.J., vol. 54, pp. 1155-1175, July 1975.
24. B.G. Haskell, et al., "Interframe Coding of Videotelephone Pictures," Proc. IEEE, vol. 60, pp. 792-800, July 1972.
25. B.G. Haskell, et al., "Interframe Coding of 525-Line, Monochrome Television at 1.5 Mbits/sec.," IEEE Trans. on Commun., vol COM-25, pp. 1339-1348, November 1977.
26. B.G. Haskell and R.L. Schmidt, "A Low Bit-rate Interframe Coder for Videotelephone," B.S.T.J., vol. 54, pp. 1475-1495, October 1975.
27. G.T. Herman, et al., "Rapid Computerized Tomography," in M. Landet, J. Anderson and S. Begon, Medical Data Processing, pp. 581-598. London: Taylor and Francis, 1976.
28. J.J.Y. Huang and P.M. Schultheiss, "Block Quantization of Correlated Gaussian Random Variables," IEEE Trans. on Commun. System., vol. 11, pp. 289-296, September 1963.
29. IEEE Trans. on Communications, special issue on Image Coding, vol. COM-25, November 1977.
30. T. Ishiguro, et al., "Composite Interframe Coding of NTSC Color Television Signals," National Telecommun. Conference, Dallas, Texas, 1976.

31. A.K. Jain, Multidimensional Techniques in Digital Image Processing, (book), to appear.
32. A.K. Jain, "A Fast Karhunen-Loeve Transform for a Class of Random Processes," IEEE Trans. on Commun., vol. COM-24, pp. 1023-1029, September 1976.
33. A.K. Jain, "A Sinusoidal Family of Unitary Transforms," to appear in IEEE Trans. on Machine Intelligence and Pattern Analysis.
34. A.K. Jain, "Partial Differential Equations and Finite Difference Methods in Image Processing - Part I: Image Representation," J. Optimization Theory and Applications, vol. 23, pp. 65-91, September 1977.
35. A.K. Jain and J.R. Jain, "Partial Differential Equations and Finite Difference Methods in Image Processing, Part II: Image Restoration," IEEE Trans. on Automatic Control, vol. AC-23, pp. 817-834, October 1978.
36. A.K. Jain, S.H. Wang, and Y.Z. Liao, "Fast Karhunen-Loeve Transform Data Comparison Studies," National Telecommun. Conference, Dallas, Texas, 1976.
37. J.R. Jain, A.K. Jain and R.A. Robb, "Data Comparison of Multidimensional X-ray Images," submitted for publication.
38. N.S. Jayant, Waveform Quantization and Coding. New York: IEEE Press, 1976.
39. S.C. Knauer, "Real-Time Video Compression Algorithm for Hadamard Transform Coding," IEEE Trans. on Electromagnetic Compatibility, vol. EMC-18, pp. 28-36, February 1976.
40. H.P. Kramer and M.V. Mathews, "A Linear Coding for Transmission of Correlated Signals," IRE Trans. on Inform. Theory, vol. IT-2, pp. 41-46, September 1956.
41. A.J. Kurtenbach and P.A. Wintz, "Quantizing for Noisy Channels," IEEE Trans. on Commun. Tech., vol. COM-17, pp. 291-302, April 1969.
42. J.O. Limb, et al., "Combining Intraframe and Frame to Frame Coding for Television," B.S.T.J., vol. 53, pp. 1137-1173, July 1974.
43. J.O. Limb and J.A. Murphy, "Measuring the Speed of Moving Objects from Television Signals," IEEE Trans. on Commun., vol. COM-23, pp. 474-478, April 1975.
44. J.O. Limb and F.R.W. Pease, "A Simple Interframe Coder for Video Telephony," B.S.T.J., vol. 50, pp. 1877-1888, August 1971.
45. S. Lin, An Introduction to Error-Correcting Codes. Englewood Cliffs: Prentice-Hall, 1970.

46. J.L. Mannons and D.J. Sakrison, "The Effects of a Visual Fidelity Criterion on the Encoding of Images," IEEE Trans. on Inform. Theory, vol. IT-20, pp. 525-536, July 1974.
47. J. Max, "Quantizing for Minimum Distortion," IRE Trans. on Inform. Theory, vol. IT-6, pp. 7-12, March 1960.
48. F.W. Mounts, "A Video Encoding System with Conditional Picture-Elements Replenishment," B.S.T.J., vol. 48, pp. 2545-2554, September 1969.
49. T.R. Natarajan and N. Ahmed, "On Interframe Transform Coding," IEEE Trans. on Commun., vol. COM-25, pp. 1323-1329, November 1977.
50. A.N. Netravali and J.D. Robbins, "Motion-Compensated Television Coding: Part I," B.S.T.J., vol. 58, pp. 631-670, March 1979.
51. T. Ohira, et al., "Orthogonal Transform Coding System for NTSC Color Television Signals," IEEE Trans. on Commun., vol. COM-26, pp. 1454-1463, October 1978.
52. J.B. O'Neal, "Predictive Quantizing Systems for the Transmission of Television Signals," B.S.T.J., vol. 45, pp. 689-721, May 1966.
53. J.B. O'Neal and T.R. Natarajan, "Coding Isotropic Images," IEEE Trans. on Inform. Theory, vol. IT-23, pp. 697-707, November 1977.
54. M.D. Paez and T.H. Glisson, "Minimum M.S.E. Quantization in Speech PCM and DPCM Systems," IEEE Trans. on Commun., vol. COM-20, pp. 225-230, April 1972.
55. A. Papoulis, Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.
56. J. Pearl, et al., "Performance Measures of Transform Data Coding," IEEE Trans. on Commun., vol. COM-20, pp. 411-415, June 1972.
57. R.F.W. Pease and J.O. Limb, "Exchange of Spatial and Temporal Resolution in Television Coding," B.S.T.J., vol. 50, pp. 191-200, January 1971.
58. W.K. Pratt, Digital Image Processing. New York: Wiley, 1978.
59. W.K. Pratt, "Correlation Techniques of Image Registration," IEEE Trans. on Aerospace and Electronic Systems, vol. AES-10, pp. 353-358, May 1974.
60. Proceedings of IEEE, special issue on Image Processing, vol. 60, July 1972.
61. L.R. Rabiner, et al., "The Chirp Z-Transform Algorithm," IEEE Trans. on Audio & Electroacoustics, vol. AU-17, pp. 86-92, June 1969.

AD-A076 841

CALIFORNIA UNIV DAVIS SIGNAL AND IMAGE PROCESSING LAB F/S 9/4
INTERFRAME ADAPTIVE DATA COMPRESSION TECHNIQUES FOR IMAGES.(U)
AUG 79 J R JAIN : A K JAIN DAAG29-78-8-0206

UNCLASSIFIED

ARO-16222.2-EL

NL

3 1/2 3

2070240



END

DATE

FORM

1-80

100

62. E.L. Ritman, et al., "Quantitative Imaging of the Structure and Function of the Heart, Lungs, and Circulation," Mayo Clinic Proc., vol. 53, pp. 3-11, January 1978.
63. R.A. Robb, "Quantitative Dynamic Three Dimensional Imaging of the Heart and Lungs by Computerized Synchronous Cylindrical Scanning Reconstruction Tomography," in P.H. Heintzon and J.H. Bürsh, Roentgen-Video-Techniques for Dynamic Studies of Structure and Function of the Heart and Circulation, 2nd International Workshop Conference, pp. 285-299, April 1976.
64. R.A. Robb, et al., "Computerized X-ray Reconstruction of Tomography in Stereometric Analysis of Cardiovascular Dynamics," Proc. Society of Photo-Optical Instrumentation Engineers, vol. 89, pp. 69-82, August 1976.
65. R.A. Robb, et al., "Three Dimensional Visualization of the Intact Thorax and Contents: A Technique for Cross-Sectional Reconstruction from Multiplanar X-ray Views," Computer and Biomedical Research, vol. 7, pp. 395-419, 1974.
66. F. Rocca and S. Zanoletti, "Bandwidth Reduction via Movement Compensation on a Model of the Random Video Process," IEEE Trans. on Commun., vol. COM-20, pp. 960-965, October 1972.
67. J.A. Roese, Interframe Coding of Digital Images Using Transform and Transform/Predictive Techniques, USCIPR Report 700, Image Processing Institute, University of Southern California, June 1976.
68. J.A. Roese, et al., "Interframe Cosine Transform Image Coding," IEEE Trans. on Commun., vol. COM-25, pp. 1329-1338, November 1977.
69. A. Rosenfeld and A.C. Kak, Digital Picture Processing. New York: Academic Press, 1976.
70. A. Segal, "Bit Allocation and Encoding for Vector Sources," IEEE Trans. Inform. Theory, vol. IT-22, pp. 162-169, March 1976.
71. R.V. Shack, "The Influence of Image Motion and Shutter Operation on Photographic Transfer Function," Applied Optics, vol. 3, pp. 1171-1181, October 1964.
72. D.K. Sharma and A.N. Netravali, "Design of Quantizers for DPCM Coding of Picture Signals," IEEE Trans. on Commun., vol. COM-25, pp. 1267-1274, November 1977.
73. R. Stamminger, "Future Use of the Geostationary Arc," Telecommunications, vol. 12, pp. 119-122, November 1978.
74. M. Tasto and P.A. Wintz, "Image Coding by Adaptive Block Quantization," IEEE Trans. on Commun., vol. COM-19, pp. 957-971, December 1971.

75. S.H. Wang, Applications of Stochastic Models for Image Data Compression, Ph.D. Thesis, Department of Electrical Engineering, State University of New York at Buffalo, 1979.
76. S. Watanabe, "Karhunen-Loeve Expansion and Factor Analysis: Theoretical Remarks and Applications," Trans. Fourth Prague Conf. Inform. Theory, Statistical Decision Functions and Random Processes, pp. 635-660, 1965.
77. P.A. Wintz, "Transform Picture Coding," Proc. IEEE, vol. 60, pp. 809-820, July 1972.
78. G.A. Wolf, The Optimum Mean Square Estimate for Decoding Binary Block Codes, Ph.D. Thesis, Department of Electrical Engineering, University of Wisconsin, Madison, 1973.
79. G.A. Wolf and R. Redinbo, "The Optimum Mean-Square Estimate for Decoding Binary Block Codes," IEEE Trans. on Inform. Theory, vol. IT-20, pp. 344-351, May 1974.
80. E. Wong, "Two Dimensional Random Fields and Representation of Images," SIAM J. of Appl. Math., vol. 16, pp. 756-770, July 1968.
81. J.W. Wood, "Two Dimensional Discrete Markovian Fields," IEEE Trans. on Inform. Theory, vol. IT-18, pp. 232-240, March 1972.
82. R.C. Wood, "On Optimum Quantization," IEEE Trans. on Inform. Theory, vol. IT-15, pp. 248-252, March 1969.
83. J.K. Yan and D.J. Sakrison, "Encoding of Images Based on a Two Component Source Model," IEEE Trans. on Commun., vol. COM-25, pp. 1315-1322, November 1977.
84. H. Yasuda, et al., "Transmitting 4-MHz TV Signals by Combinational Difference Coding," IEEE Trans. on Commun., vol. COM-25, pp. 508-516, May 1977.

ADDITIONAL REFERENCES

85. A.K. Jain and J.R. Jain, Radar Image Modeling and Processing for Real-Time RF Simulation, Final Report, ARO Grant No. DAAG2977G0044, Department of Electrical Engineering, State University of New York at Buffalo, February 1978.
86. A.K. Jain and S.H. Wang, Stochastic Image Models and Hybrid Coding, Final Report, Contract No. N00953-77-C-003 MJE, Department of Electrical Engineering, State University of New York at Buffalo, October 1977.
87. W.K. Pratt, et al., "Slant Transform Image Coding," IEEE Trans. on Commun., vol. COM-22, pp. 1075-1093, August 1974.

88. R.J. Arguello, H.R. Sellner, and J.R. Stuller, "The Effect of Channel Errors in the Differential Pulse-Code Modulation Transmission of Sampled Imagery," IEEE Trans. on Commun. Tech., vol. COM-19, pp. 926-933, December 1971.
89. B. Fox, "Discrete Optimization via Marginal Analysis," Management Science, vol. 13, pp. 201-216, November 1966.
90. D.E. Knuth, Searching and Sorting, vol. 3, The Art of Computer Programming. Reading, Massachusetts: Addison Wesley, 1973.